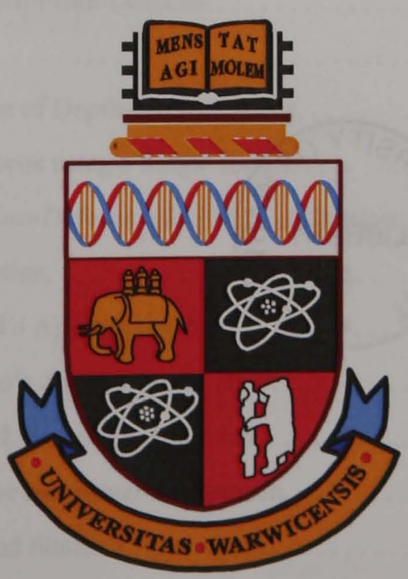
Colour Depth-From-Defocus Incorporating Experimental Point Spread Function Measurements

by

Christopher David Claxton

A thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

School of Engineering
University of Warwick



January 2007

Table of Contents

Table of Figures

Chapter 4

Chapter 6

Chapter 7

Table of Tables

Chapter 7

Appendix C

Appendix E

Acknowlegdements

First and foremost, I would like to thank my research supervisor Dr. Richard Staunton for all his guidance and support over the last four years.

I would like to thank Dr. Geoff Diamond for his encouragement, stimulating discussions and practical electronic help.

Photographic knowledge and expertise was very gratefully received from Dr. Ir. Jan Rakels.

I am grateful to Huw Edwards and Stephen Wallace for their work in constructing the gantry of the x-stage.

Dr. Rick Chartrand at Los Alamos National Laboratories provided a valuable MATLAB implementation of the regularised numerical differentiation, which was much appreciated.

The University of Warwick kindly funded this research in the form of the Warwick Postgraduate Research Fellowship and I'm very thankful for the privilege of being able to study for this PhD and for the valuable opportunities it afforded me.

Dedication

To my parents, Michael and Melanie Claxton, for their efforts, support, encouragement, kindness and generosity over the years and without whom this thesis would never have been possible.

To my grandparents, Cecil and June Claxton, for inspiring me to be creative as a child through drawing, woodwork, building and sailing yachts, playing with Meccano and electrical circuits. The days spent in the shed are some of my fondest memories.

And to Ruth Esther Morwenna (Hawken), who is truely a wonderful and specially chosen gift from God.

Declaration

This thesis is submitted in partial fulfilment for the degree of Doctor of Philosophy under the regulations set out by the Graduate School at the University of Warwick.

This thesis is solely composed of research completed by Christopher David Claxton under the supervision of Dr. Richard Staunton.

None of the work presented here has been published or submitted for another degree.

Abstract

Depth-From-Defocus (DFD) is a monocular computer vision technique for creating depth maps from two images taken on the same optical axis with different intrinsic camera parameters. A pre-processing stage for optimally converting colour images to monochrome using a linear combination of the colour planes has been shown to improve the accuracy of the depth map. It was found that the first component formed using Principal Component Analysis (PCA) and a technique to maximise the signal-to-noise ratio (SNR) performed better than using an equal weighting of the colour planes with an additive noise model. When the noise is non-isotropic the Mean Square Error (MSE) of the depth map by maximising the SNR was improved by 7.8 times compared to an equal weighting and 1.9 compared to PCA. The fractal dimension (FD) of a monochrome image gives a measure of its roughness and an algorithm was devised to maximise its FD through colour mixing. The formulation using a fractional Brownian motion (fBm) model reduced the SNR and thus produced depth maps that were less accurate than using PCA or an equal weighting. An active DFD algorithm to reduce the image overlap problem has been developed, called Localisation through Colour Mixing (LCM), that uses a projected colour pattern. Simulation results showed that LCM produces a MSE 9.4 times lower than equal weighting and 2.2 times lower than PCA.

The Point Spread Function (PSF) of a camera system models how a point source of light is imaged. For depth maps to be accurately created using DFD a high-precision PSF must be known. Improvements to a sub-sampled, knife-edge based technique are presented that account for non-uniform illumination of the light box and this reduced the MSE by 25%. The Generalised Gaussian is presented as a model of the PSF and shown to be up to 16 times better than the conventional models of the Gaussian and pillbox.

Abbreviations

ADC	Analogue-to-Digital Converter
APS	Active Pixel Sensors
AR	Auto-Regressive
AWGN	Additive White Gaussian Noise
CCD	Charge-Coupled Device
CFA	Colour Filter Array
CMOS	Complementary Metal Oxide Semiconductor
CRT	Cathode Ray Tube
CTE	Charge Transfer Efficiency
CTF	Charge Transfer Function
DFD	Depth-From-Defocus
DFF	Depth-From-Focus
DFMB	Depth-From-Motion-Blur
DFT	Discrete Fourier Transform
DOF	Depth-Of-Field
fBm	fractional Brownian motion
FD	Fractal Dimension
FPA	Focal Plane Array
FPN	Fixed Pattern Noise
FPS	Frames Per Second
FT	Fourier Transform
GA	Genetic Algorithm
HSI	Hue-Saturation-Intensity
IC	Integrated Circuit
LCM	Localisation through Colour Mixing
ML	Maximum Likelihood
MRF	Markov Random Field
MSE	Mean Square Error
MTF	Modulation Transfer Function

ND	Neutral Density
n-D	N-dimensional
OTF	Optical Transfer Function
PCA	Principal Component Analysis
PDF	Probability Density Function
PRF	Pixel Respose Function
PSF	Point Spread Function
RGB	Red-Green-Blue
RMS	Root Mean Square
RND	Regularised Numerical Differentiation
SLR	Single Lens Reflex
SNR	Signal-to-Noise Ratio
STFT	Short-Time Fourier Transform

Chapter 1

Introduction

1.1 Computer Vision

Computer vision is a fascinating marriage of mathematics, physics, computer programming, engineering and to some extent biology. Physics describes how photons are emitted by light sources, are reflected and refracted by objects in a scene and the effect they have when they impinge on a photo-sensor, such as a CCD. Mathematics provides the universe in which to perform calculations. In particular, Fourier transform theory is highly used in engineering and computer vision is no exception. Computer programming provides an efficient way of performing the repetitive calculations. Creativity and inventiveness lies in the design of algorithms to best utilise the information obtained about a scene and biological counterparts can be the inspiration behind techniques.

1.2 Optical Lenses

1.2.1 Historical Perspective

The leading theory of the creation of the Universe, as researched by cosmologists, suggests that space and time began around 13.7 billion years ago, and with it came the creation of photons. Around 4.6 billion years ago our solar system and the Earth were formed and 600 million years later the first primitive life appeared. It took about another 50 million years for the first light sensitive cells to evolve. With the dinosaurs extinct, Homo sapiens were living in Africa about 100,000 years ago. The complex vision exhibited by humans enabled them to hunt, make tools, kindle fires, and more. Colour cave paintings in France and Spain during the early stone age of 13,000 BC show that man had developed artistic talents to pictorially represent their lives as well as the concept of numbers.

Volcanoes form naturally occurring glass called *obsidian* that was used during the Stone Age to produce sharp blades or arrowheads and possibly early mirrors. It can be ground to produce blades much sharper than high quality steel and it is now used in cardiac surgery.

Glass is thought to have been made during the Bronze Age around 3000 BC in Mesopotamia, which was the same time that the first written alphabets were developed. The first written records of the use of glass to focus the sun's rays to ignite a material are recorded in Aristophanes' play *The Clouds* written around 400BC [1]:

Strepsiades: Have you ever seen this stone in the chemist's
shops, the beautiful and transparent one, from which
they kindle fire?

Socrates: Do you mean the burning-glass?

Strepsiades: I do.

In around 40 AD it is known that the Roman statesman and writer Seneca used a glass container filled with water to magnify text [2]. By 1000 AD the first *reading stone* was invented to magnify text solely using glass and then in 1200 AD in Italy the first eyeglasses were created. The ability of a carefully shaped piece of glass to magnify small objects, compress a large object into a small volume, focus the sun's rays and thus begin combustion in a material and allow a reader to see more clearly surely makes it an ancient marvel.

1.2.2 Defocusing

Light travels at 186,000 miles per second and its path is altered by refraction, diffraction and reflection (and gravity). A convex (converging) lens produces a focused image using the laws of refraction and the relationship between the object and the image is given by the Gaussian lens law that states

$$\frac{1}{F} = \frac{1}{u} + \frac{1}{v} \tag{1.1}$$

where F is the focal length of the lens and u and v are the distances of the object and the image plane from the lens respectively. If the image plane is not at the correct distance for a given object then the image produced by the lens is defocused. Each point on the object is imaged to a non-point-like shape that is determined by the shape of the aperture, diffraction effects, the distance of the object and the wavelengths of light being reflected. For a circular aperture each point becomes a blur circle, assuming geometrical optics. It can be shown that the radius of the blur circle σ for an object a distance D from the lens is given by

$$\sigma = \frac{vD - F(v+D)}{fD} \tag{1.2}$$

where F is the focal length of the lens, f is the nominal f-number defined as $f = \frac{F}{2r}$, r is the radius of the aperture, v is the distance between the lens and image plane (e.g. CCD) [3]. A given blur circle radius can be produced by two points, one either side of the focus position of the lens. If the blur circle radius can be measured then the depth of the point can be found from re-arranging (1.2) to give

$$D = \frac{Fv}{v - F - \sigma f} \tag{1.3}$$

assuming the object is further from the lens than the focus position.

The defocus effect acts as a spatial low-pass filter in 2D and for a non-light absorbing lens the volume of the PSF is unity [4]. There are two models of the PSF due to defocus that appear frequently in papers: one of them is the cylindrical or pillbox PSF given by

$$h(x, y) = \begin{cases} \frac{1}{\pi r^2} & \sqrt{x^2 + y^2} \leq r \\ 0 & \text{otherwise} \end{cases} \tag{1.4}$$

where r is the radius of the pillbox; and the second is the 2D Gaussian given by

$$h(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right\} \tag{1.5}$$

where σ_x and σ_y are the standard deviations of the Gaussian in the x and y directions respectively. Note that the blurring due to the Gaussian PSF leads to an exponential decay in the frequency domain [5]. Both PSFs are assumed to be centred on (0, 0) so that there is no phase shift due to defocusing, which is a valid assumption for a well-centred optical system.

The pillbox model is the mathematically obtained PSF shape assuming geometrical optics (i.e. neglecting the wave-nature of light) and a circular aperture. The Gaussian PSF has been proposed as a suitable practical model because it approximates the effects of diffraction, polychromatic light [6], sampling by a CCD and low-pass filtering by the camera electronics.

Bove [7] noted that in order to model the image captured by a lens, the Huygens-Fresnel integral for Fraunhofer diffraction would have to be applied to each image point. This is inherently complex and a simpler approach is to consider a small region in which it is assumed that the depth is constant. This allows the space-variant blurring problem to be approximated as a space-invariant problem, thus allowing convolutions to be employed, which are much easier mathematically. This strategy suffers from the trade off of precision though [8].

The problem with changing the distance between the image plane (CCD) and the lens is that an undesirable change of magnification occurs, as shown in Figure 1.1. If the sensor plane is at position I_F then the object is in focus. Moving the sensor to either I_1 or I_2 results in a blur circle with a centre that moves along the axis of the principal ray, which passes through the centre of the lens. The lack of registration can be improved by using a combination of zooming and focusing [9], but this requires expensive computer-controlled lenses and calibration [10]. Image registration and warping could be performed in software [11], but accurate interpolation is essential and this is an extra computational overhead.

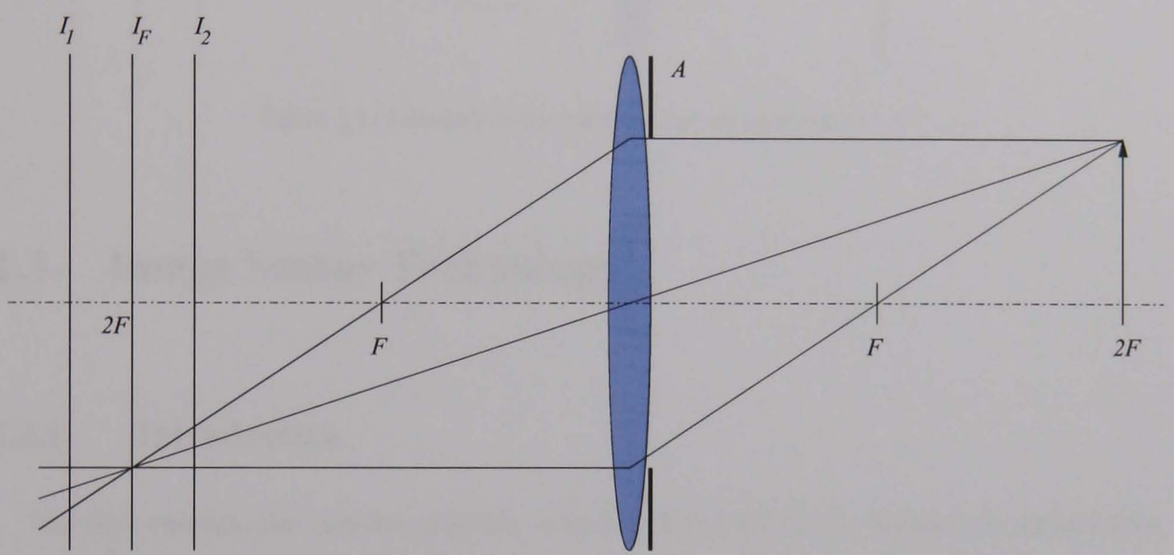


Figure 1.1: Conventional lens model with an aperture A and a focal length F

The addition of an aperture A' at a distance of the focal length F in front of the lens on the object side, to form a *telecentric* lens, theoretically eliminates the problem [12] but practically it can be used to reduce the shifts to less than 0.1 pixels [10]. From geometrical optics it is known that a ray that passes through a point the distance of the focal length in front of the principal plane is refracted by the lens to be parallel to the optical axis, as shown in Figure 1.2, thus removing the registration problem. An alternative solution is to employ a convex lens in between the last lens element and the sensor plane to make the principal ray parallel to the optical axis, but this changes the properties of the lens [10]. Watanabe and Nayar [13] showed that ignoring the magnification problem can result in significant errors in the depth map using DFD.

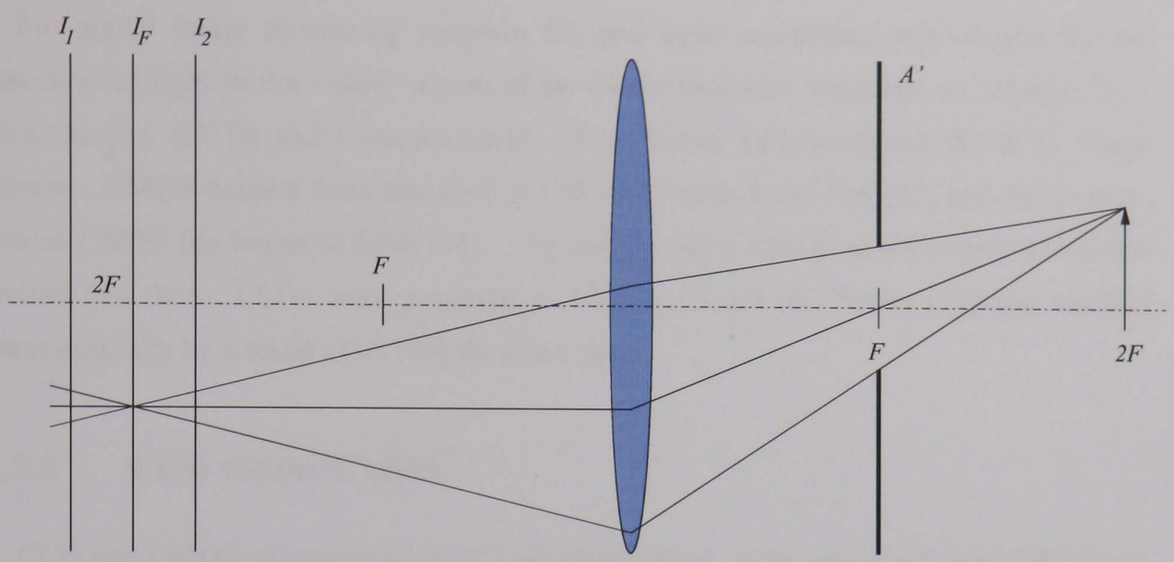


Figure 1.2: Telecentric lens model with an external aperture A'

1.3 Image Sensor Technology

1.3.1 Introduction

The first camera, the *camera obscura*, which is Latin for *dark chamber*, is believed to have been accidentally discovered by Alhazen in the 10th century. It did not employ a lens, but had a pinhole (a very small aperture) that produces an image of the scene that lacks depth information [14], but retains brightness information. The camera was essentially a dark room with a pinhole on one side and paper on the other and by about the 15th century it was being used by artists, who would trace the inverted image produced and thus produce images with very good perspective.

The first permanent photograph was created by Joseph Nicéphore Niépce (1765-1833) and the very long exposure time of between 8 and 20 hours restricted his work to architecture.

In a conventional SLR camera, a lens, often composed of many elements, is used to bring the scene into focus and the image is projected through an aperture and onto a mirror that directs the light towards the viewfinder. To take a photo, the shutter button is pressed, which drops the mirror and allows the light to travel through the mechanical shutter and onto the film. The light causes a chemical reaction that results in a negative brightness image. After the set exposure time has elapsed the shutter closes and the mirror slides back into position.

For digital image processing purposes the two main competing technologies for the detection of light in the visible region of the electromagnetic spectrum are Charge-Coupled Devices (CCD) and Complementary Metal Oxide Semiconductor (CMOS) image sensors. CMOS circuits were invented in 1963 by Wanlass and Sah [15] and the production of CMOS ICs began in 1968 [16]. The early passive CMOS sensors were developed around this time. CCDs were proposed in 1970 by Boyle and Smith [17] and verified experimentally by Amelio *et al.* [18] the same year.

1.3.2 CCD versus CMOS

CCD and CMOS sensors consist of a pixelated metal oxide semiconductor (MOS) and function using the photoelectric effect, which is where an incident photon generates an electron-hole pair and the photoelectron is then stored.

The two major pixel designs are *photogates* and *photodiodes*, schematics of which are shown in Figure 1.3. A photogate is a MOS capacitor that stores photoelectrons in a voltage-induced potential well [19]. The polysilicon gate over the pixel reduces its sensitivity, especially to wavelengths in the blue end of the visible spectrum, but all of the pixel is photosensitive. A photodiode is created by ion implantation and photoelectrons are stored in the depletion region around the p-n junction [19]. The photogate has a higher full well capacity and thus it possesses a higher dynamic range because it can handle a wider range of illumination intensities [19]. The fill factor of a photodiode can be improved using a microlens and the reduced sensitivity of photogates can be improved using thin polysilicon gates.

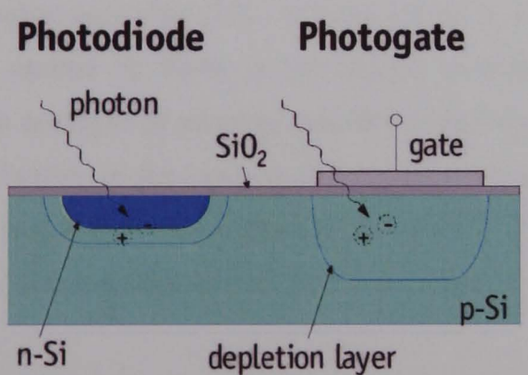


Figure 1.3: Schematic of photodiode and photogate technologies (from [19])

Quantum efficiency is the ratio of the collected electrons to the number of incident photons and ideally it is 100% for all wavelengths in the visible region between 440nm and 700nm. Losses due to absorption, reflection and transmission must be minimised by manufacturers to approximate the ideal behaviour. Absorption losses occur due to optically dead regions [20] and they are greater in CMOS devices due to the presence of

readout transistors. Transmission losses occur when a photon passes straight through the photosensitive volume without generating a photoelectron and it is most pronounced at very short and very long wavelengths. The wavelength-dependent reflection of silicon causes reflection losses, but it can be reduced using anti-reflective coatings [20]. Better quantum efficiencies can be achieved by thinning the obstructing layers and using back-side illumination (for which the quantum efficiency can be about 90% for CCDs), instead of usual front-side illumination, however, this requires more complex manufacturing processes.

The dynamic range of a sensor is the ratio of the device's output at saturation to its noise floor. If the light intensity is too high for a given exposure time, saturation occurs. Blooming is the effect where the excess charge spills over to other pixels. CMOS pixel sensors typically have a drain to absorb charge overflow and thus has natural anti-blooming, unlike CCDs [19].

The Charge Collection Efficiency (CCE) measures the ability of a pixel to retain its photoelectrons and it is important because it affects the spatial resolution. CCD pixels have a higher electric field than CMOS pixels and so diffusion effects are almost eliminated compared to CMOS sensors [21]. The unwanted thermal diffusion effects make the resulting image look defocused.

CMOS sensors allow random access as they are directly addressable, whereas the pixels of CCDs must be read out in a fixed sequence. Thus, windowing can be achieved on CMOS sensors leading to higher frame rates for a reduced image size, and this is not possible for CCDs. CMOS sensors have a charge-to-voltage conversion in each pixel whereas CCDs have one per array, thus the charge packets must travel a long way through the silicon to reach the output amplifier [21]. It is important in CCDs that the channel is devoid of electron traps caused by flaws in the design, manufacturing process or the silicon itself. The on-chip analogue processing circuitry means that CMOS detectors have higher noise levels. Variations in the open-loop amplifiers caused by wafer processing variations mean that the response of each pixel under uniform illumination is different. This non-uniformity is a disadvantage and it has been improved using feedback-based amplifiers.

In conclusion, CCDs are still the technology of choice for scientific applications [22] as they have very good image quality, low dark current, high quantum efficiency and a high fill-factor [23]. The lack of random access, higher power consumption and larger clock driver voltages of CCDs [22] are generally not important in machine vision.

1.3.3 CCD Architectures

Over the years different CCD architectures have been developed, some of which are illustrated in Figure 1.4. Photons impinge on the light-sensitive region and a fraction of those are converted to photoelectrons through the photoelectric effect. A *progressive scan* readout process is then used where the charge is shifted row by row into the serial readout register. Each packet of charge is then converted to a voltage. A problem with this architecture is *charge smearing* caused by light falling on the sensor during the readout process. However, a mechanical shutter or stroboscopic light source can alleviate the problem [24]. Mechanical shutters are relatively slow and also have limited lifetimes [25]. The full-frame CCD is the simplest to manufacture and operate and it gives the highest resolution for a given chip size of the architectures reviewed [24]. It is generally based on photogate technology [19].

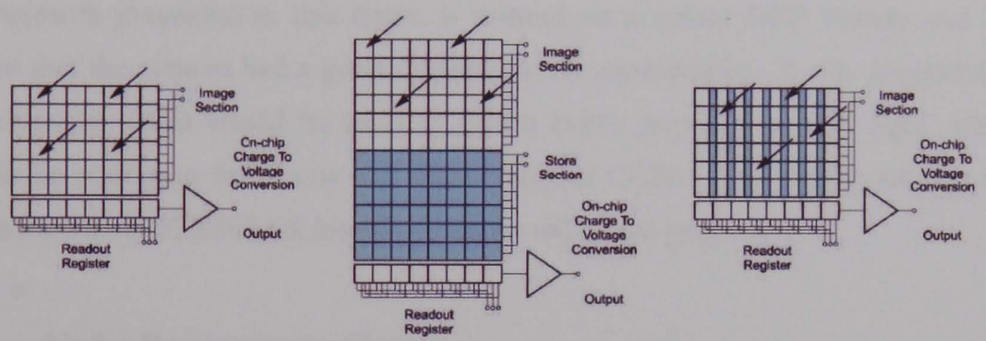


Figure 1.4: Full-frame (left), frame-transfer (middle) and interline-transfer (right) CCD schematics (from [25])

The frame-transfer CCD uses a two part sensor, one of which is photosensitive (and also usually based on photogates) and the other is protected by a light-tight mask and is known as the storage region [25]. The photoelectrons generated by the imaging region are transferred to the storage region at high speed and from the storage region, the charge packets are read out. This architecture was designed to help alleviate the charge smearing problem with full-frame CCDs and so a mechanical shutter or stroboscopic light source is not required [24]. Faster frame rates and continuous light collecting are achievable with frame-transfer CCDs, however, charge smearing is not completely removed and twice the amount of silicon is required due to the storage region, which increases the sensor size and cost [25].

Charge smear is virtually eliminated in interline-transfer CCDs that incorporate charge transfer channels called *interline masks*. The channels allow the charge to be rapidly shifted, but there is still some smear due to *light-piping*, which is where scattered and reflected photons move to the shift register. This problem has been further reduced using frame-interline-transfer CCDs that possess a storage region like a frame-transfer CCD

[24]. Interline-transfer CCDs are generally based on photodiodes instead of photogates and electronic shuttering can be achieved by altering the voltages at the photodiode so that photoelectrons are injected into the substrate [25]. The interline mask reduces the light sensitive area of the sensor and thus reduces the fill-factor, however, this can be remedied to some extent using a microlens array. The advantage of using photodiodes compared to photogates is better sensitivity to blue light, but they have a lower dynamic range than full-frame CCDs due to the lower fill-factor [19].

For scientific vision systems, CCDs are the best of the two alternative technologies. Both the frame-transfer and the interline-transfer architectures have their merits. The advantage of a frame-transfer approach is the higher fill factor, leading to an increased dynamic range; however, charge smearing and poor response to blue light remains a problem. In contrast, the interline-transfer design has a better response to blue light and charge smearing is virtually eliminated, but at the cost of a smaller dynamic range due to the interline mask taking up silicon.

The research presented in this thesis is centred on a colour DFD system and it was important that the camera had a good response to all wavelengths. It was decided that an interline-transfer CCD would be used to give a better response to blue light, which is known to be poorer in full-frame and frame-transfer CCDs. The Basler A631fc colour camera with a Sony ICX267AK interline-transfer CCD was employed.

1.3.4 Noise Processes in Cameras

Signals in an imaging system can be divided into wanted and unwanted, where the latter category is generally termed noise. The CMOS and CCD sensors have sources of noise including:

- **Dark noise**: The atoms in the sensor vibrate and occasionally release a free electron from the semiconductor, which is indistinguishable from a photoelectron. It is these vibrations that lead to dark noise. The dark noise accumulates over time and if the exposure time is doubled it is expected that the noise will double. Raising the temperature of the imaging sensor results in an increase in the dark noise and so often the sensors used for astronomical imaging are cooled.
- **Readout noise**: The charge in a photosite must be measured and converted to a digital value and this requires amplification, which adds noise due to thermal oscillations. Further, the amplifier before the ADC has a built-in offset or bias. Quantisation effects due to the finite number of digital levels adds noise too.
- **Fixed Pattern Noise (FPN)**: Electronic sources, such as clock signal breakthrough and crosstalk in the array, produce FPN. It is noticeable on CMOS sensors, but it is often

negligible for CCDs [24]. It is usually independent of the integration time and temperature and it is different for each pixel.

- **Photon noise**: Photons reflected from a surface or emitted from a source will not arrive at the sensor at the same time and it is the difference in arrival times that leads to photon noise. The noise will have a Poisson distribution. An ideal, evenly illuminated surface imaged in the presence of photon noise only will show fluctuations in intensity. The effect of photon noise is increased if a fast shutter speed is employed, a dimly lit subject is being imaged or high amplification gains are required. Thus, a long exposure time should be employed. If there are many photons the Poisson distribution approximates a Gaussian distribution with a standard deviation that is dependent on the light intensity.
- **Random noise**: Electromagnetic interference, fluctuations in the power supply to the camera and electronic noise will produce random noise.
- **Cosmic ray effects**: High energy particles will leave a hot pixel or a spurious streak on an image, but these can be generally ignored, except for astronomical imaging.
- **Variations in photosite sensitivity**: Pixels on an imaging sensor are designed to be identical, but manufacturing processes and variations in the materials lead to sensors with differing sensitivities (as shown by the PRF description in Section 3.2.1). Bad pixels produce an output that is independent of the number of photons that strike the active area of the sensor. Hot pixels are permanently high and cold pixels permanently low and both are a result of manufacturing problems.
- **Dust on the sensor**: A dirty sensor that has accumulated dust will have systematic errors due to shadows, and naturally these effects can be reduced by cleaning with an appropriate solvent.

Noise is a stochastic process from which information can be gleaned from *bias frame* and *dark frame* measurements. The sensor architecture for the camera used in the research is represented in Figure 1.5. If the shutter is not opened (i.e. the minimum exposure time is employed) then the output of the camera is called the bias frame and the noise is that of the readout electronics, shown as the ADC and Variable-Gain Control (VGC) in the figure. The dark frame is taken with the maximum exposure time and the lens cap on, so that the noise is predominantly due to thermal excitation of the photosites.

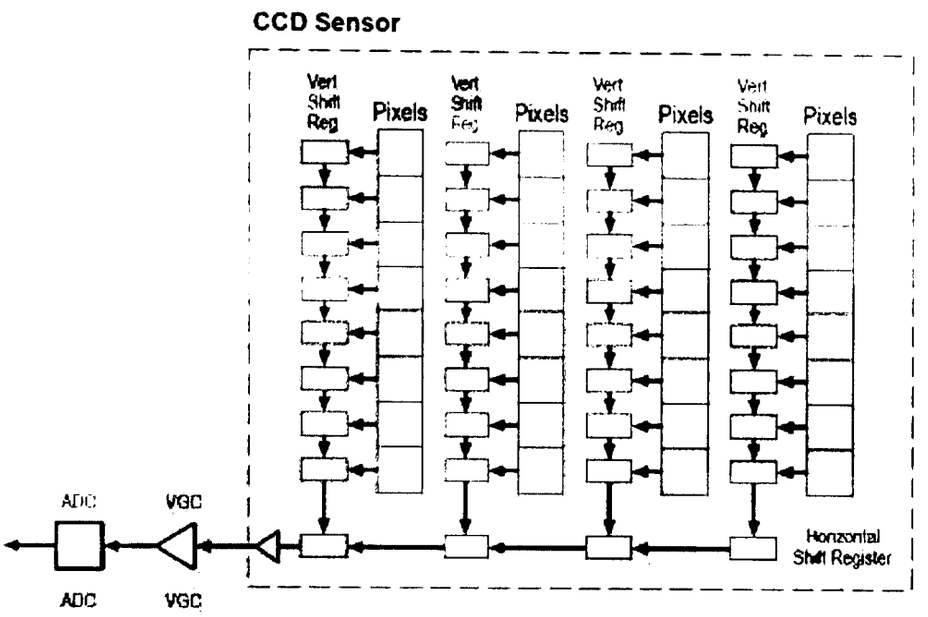


Figure 1.5: Sensor architecture (from the Basler A631fc manual [26])

1.3.5 Colour Imaging

Biological Colour Vision in Humans

In low-level vision in humans, colour helps to segment the retinal image and in high-level vision it aids recognition of objects with shape information obtained from edges. Impressionist artists such as Monet painted images that deliberately lack accurate object shapes but the viewer quickly recognises the subjects due to the careful use of colour [27].

Light is focused by the cornea and lens to form an image on the retina [28]. The retina of the human eye is circular and approximately 42mm in diameter [29] and it is divided into two regions, namely the central and peripheral regions. The two types of photodetectors are rods and cones and the latter are subdivided into three types according to their spectral frequency response. The peak sensitivities of the detectors are 430nm, 530nm and 560nm and as they do not exactly correspond to blue, green and red respectively the terms short, medium and long wavelengths are sometimes used [30]. The brain demosaics the responses of the cones to produce colour vision. The central region is dominated by cones and that of the peripheral region by rods. The foveal pit is a region with no rods and the cones are packed as tightly as possible in a hexagonal arrangement. The fovea is about 0.5mm in diameter and of the 5 million cones in the eye about 10,000 are located in the fovea. A human who suffers from colour blindness has a deficiency in one or more types of cones and thus their eyes do not sample the visible region of the electromagnetic spectrum sufficiently.

There are about 100 million rods in the human eye and their higher quantum efficiency compared to cones outweighs their disadvantage of a lower spatial resolution as they enable humans to see in low light levels [31].

Retinal sensing is based on trichromatic principles owing to the three types of cones and the opponent colour encoding is used by the neural pathways to the brain. The opponent colours are red-green and yellow-blue, which is supported by studies that humans do not see a yellow-blue or reddish-green colours [28].

Electronic Colour Imaging

Colour images can be captured using a monochrome camera with a spinning red, green and blue colour filter in front [30]. Alternatively the colour filters could be fixed and a Colour Filter Array (CFA) employed, with the Bayer filter being one of the most common. Each set of 2×2 pixels has two green, one red and one blue filter and interpolation is used to fill in the missing values. Some cameras use the secondary colours cyan, yellow and magenta as they are better in darker conditions because less light is attenuated [32]. A more expensive approach is to employ a 3-CCD camera that uses two beamsplitters and three colour filters to capture three images in different spectral bands on the three separate CCDs.

The spectral response of the Sony ICX267AK interline-transfer CCD employed in the Basler A631fc colour camera used in the research is presented in Figure 1.6. Note in particular the reduced sensitivity to blue light, which is because the shorter wavelengths are particularly well absorbed by the polysilicon gate structures.

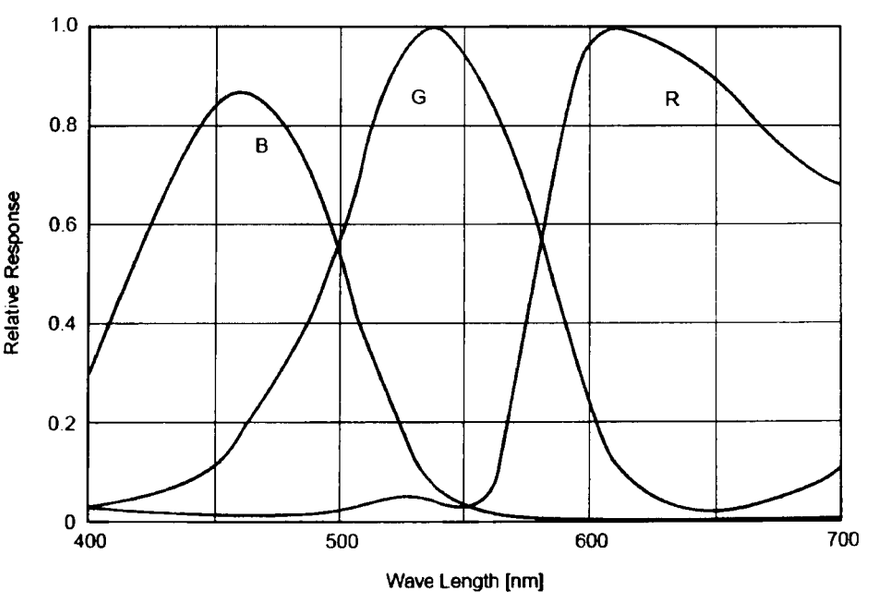


Figure 1.6: Spectral response of the Sony ICX267AK CCD (from [33])

Colour Spaces

The images are often captured in RGB format and they can be transformed to other colour spaces. Due to the subtractive nature of inks, the CMY colour space is useful for printers and it is formed from the RGB components using

$$\begin{pmatrix} C \\ M \\ Y \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} - \begin{pmatrix} R \\ G \\ B \end{pmatrix}. \tag{1.6}$$

A camera system such as a video camera captures an image and the output is proportional to the light radiated by the objects being filmed. A Cathode Ray Tube (CRT) has an output intensity that is not a linear function of the voltage and so the camera is *gamma corrected* to compensate. The R′ G′ B′ components are the gamma corrected RGB values.

The YIQ colour space used in NTSC television represents the luminance (Y), hue (I) and saturation (Q) information and it is formed from the non-linear RGB components using [34]

$$\begin{pmatrix} Y \\ I \\ Q \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{pmatrix} \begin{pmatrix} R' \\ G' \\ B' \end{pmatrix}. \tag{1.7}$$

The CMYK and RGB colour spaces are non-uniform in that the Euclidean distance between two points does not correspond to the perceptual difference between the colours. Examples of a perceptually uniform colour spaces are CIE L*a*b* (CIE LAB) and CIE L*u*v* (CIE LUV), where the former is mainly used in displays and the latter in colour imaging and printing [28]. They are formed from non-linear combinations of the RGB components. The CIE LAB space incorporates opponent colour encoding so that a* and b* correspond to the opponent hues red-green and yellow-blue respectively and L* corresponds to the lightness.

Colour information is useful in many image processing tasks including object recognition, content-based image retrieval, image compression [30], forensic image processing [35] and the detection of cancer cells [36].

1.3.6 The Representation of Digital Images

A monochrome digital image with M rows and N columns may be expressed in matrix form as

$$\mathbf{f} = \begin{pmatrix} f(0,0) & f(0,1) & f(0,2) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & f(1,2) & \cdots & f(1,N-1) \\ f(2,0) & f(2,1) & f(2,2) & \cdots & f(2,N-1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & f(M-1,2) & \cdots & f(M-1,N-1) \end{pmatrix} \tag{1.8}$$

where $f(x, y)$ represents the pixel value, and thus brightness, at spatial location (x, y). Usually $0 \leq f(x, y) \leq G-1$ where G is the number of grey levels and it is often a power of two. Many image processing tasks are performed on a small window of the image. If a 3×3 window is applied to image $\mathbf{f}$ that is centred on the pixel $f(1, 1)$ then the resulting windowed image $\mathbf{f}_w$ is given by

$$\mathbf{f}_w = \begin{pmatrix} f(0,0) & f(0,1) & f(0,2) \\ f(1,0) & f(1,1) & f(1,2) \\ f(2,0) & f(2,1) & f(2,2) \end{pmatrix}. \tag{1.9}$$

It is convenient to represent the original image $\mathbf{f}$ as a column-stacked vector $\mathbf{f}_S$ in the form

$$\mathbf{f}_S = \begin{pmatrix} f(0,0) \\ f(1,0) \\ \vdots \\ f(M-1,0) \\ f(0,1) \\ f(1,1) \\ \vdots \\ f(M-1,1) \\ \vdots \\ f(0,N-1) \\ f(1,N-1) \\ \vdots \\ f(M-1,N-1) \end{pmatrix} \tag{1.10}$$

An RGB colour image will be composed of three brightness matrices with one for each colour plane.

1.4 Methods of Capturing 3D Images

1.4.1 Introduction

Humans use a variety of vision-based depth cues including [37]:

- Texture
- Edges
- Size perspective
- Binocular disparity
- Motion parallax
- Occlusion effects
- Variations in shading
- Defocus effects

It is these cues that have provided computer vision researchers with ideas as to how to imbue computers connected to cameras with the ability to measure depth. Often the goal of 3D computer vision systems is to create a *depth map*, which is essentially a representation of the distance between the camera and points in the scene. One way to show it is as a greyscale image where brightness is directly proportional to depth.

The techniques can be divided into monocular methods and multiple view methods, which correspond to whether the intrinsic or extrinsic camera parameters are changed respectively. The intrinsic camera parameters are focal length, f-number and the distance between the image plane (CCD) and the lens. The extrinsic camera parameters are the location and orientation of the camera.

Leonardo da Vinci realised that any illuminated body reflects an infinite number of images of itself where each image is from a slightly different viewpoint and this is the basis of the multiple view methods [38]. Reconstruction of 3D scenes using stereo disparity, multiple view points and structure-from-motion approaches are multiple view methods. The monocular approaches include texture gradient analysis, occlusion cues and depth-from-focus and defocus, the latter being the centre of this research. Two images captured with different shutter speeds of the same moving scene provides depth information and algorithms that are based on this principle are called depth-from-motion blur [39].

Owls improve their depth perception, and thus locate their prey better, even though sitting otherwise still by moving their heads from side-to-side, and so employing motion parallax [31]. A moving observer obtains a lot of 3D information by traversing through a scene. However, the notion that the brain progressively builds up an image of the world with each fixation is challenged by visual tests that show a phenomenon known as *change blindness* [40] where an observer can fail to notice large changes in two images when a grey image is presented in between the images of interest. The brain appears to create a limited description of the scene and then simply ignores the rest of the information presented by the eyes.

1.4.2 Depth maps

Stereopsis is the process of combining a pair of 2D brightness images in order to recover depth information using triangulation and it is exhibited naturally by humans. Occlusions, limited fields-of-view and parts of the image with uniform colour or intensity make determining the required correspondences in the image difficult [41].

Depth-from-focus camera system change the camera parameters to maximise the focus of a scene, usually measured by maximising the energy [42] or sharpness (such as the Tenengrad, variance or sum-modified-Laplacian [43]) or in the frequency domain, using high frequency measures [11]. From the known camera parameters, the depth can be calculated using the Gaussian lens law and interpolation can be used to increase the depth resolution [43]. The problem with this approach is that usually many images are required, for example between eight and thirty [14], and this places strict requirements on the static nature of the scene. Changing the focal length or the distance between the lens and image plane produces magnification effects that must be removed using either image warping in software following a calibration phase [44] or using a telecentric lens. Although the technique requires only one camera, it can produce absolute depth maps and does not suffer from occlusion problems [41].

The depth-of-field specifies the range of distances from the camera for which an object placed in the range appears in focus on the image. Essentially, the blur circle radius is too small to be detected by the sensor and thus it produces a sharp image. The limited depth-of-field may sound problematic, but in fact depth-from-defocus (DFD) systems rely on this property of a lens to infer the depth of objects. Two images taken with different, known focus settings can be analysed to examine the relative blurring between the images, from which the depth of the object can be calculated. As the Literature Review in Chapter 2 shows, the mean depth range of the algorithms examined is 1.1m to 1.8m and so it is generally restricts the applications it can be applied to. A frequent assumption in the

algorithms is that the depth is constant and thus to ensure the depth returned is accurate, a small window is required, but for reliability in the presence of noise, a large window is required.

Surface orientation can be extracted from a single image using shape-from-shading techniques, but the task is cumbersome and requires very controlled environments [45]. Further, the technique cannot recover absolute depth information and thus must be combined with stereo approaches, for example, to recover absolute depth.

The relative motion between the observer and stationary objects provides important information about the shape and depth of the scene [45]. The parallax effect observed by motion of the camera (observer) or objects can be used to determine the depth by using image point correspondences [41]. Multiple cameras are required to produce absolute depth information.

One of the depth cues presented in the list above is size perspective, which is where humans use their knowledge of the size of familiar objects. Torralba and Oliva [46] developed an algorithm for estimating the mean depth of a scene based on the structure of the scene, which can be broken down into the global configuration, the size of the surfaces, and the textures present.

1.4.3 Volumetric Imaging

In contrast to producing depth maps, volumetric imaging allows the interior of objects to be analysed, a couple of techniques being Computerised Tomography (CT) and Magnetic Resonance Imaging (MRI). CT uses X-rays to produce volumetric information about a patient for the purposes of medical diagnostics. The different attenuation properties of the tissues allows doctors to see tumours, for example, and surgeons can visually prepare the operation. MRI is a technique that does not employ harmful ionising radiation, but instead uses a magnetic field and radio wave pulses.

1.4.4 The Applications of 3D Imaging Systems

Three-dimensional shape information is useful in a wide range of fields and examples of which are presented below. As the technology for 3D capture becomes faster and cheaper there is no doubt it will be used in an increasing number of applications.

Virtual Production

Computer Generated Imagery (CGI) has been common-place in the film industry for many years, but due to its high cost and labour intensive process it has remained out of the realm of television programme production until recently. Coupled with the fact that consumers desire more choice of programmes and channels, the use of virtual elements including scenes, actors (known as *avatars*) dressed in highly realistic clothing models, and props is under research. Virtual production is concerned with integrating virtual elements and real footage. In order to create a realistic looking scene with optical interactions (such as occlusions, shadows and reflections), the correct camera perspective and depth perception, 3D models are required of both the real and virtual elements [47]. Virtual elements can be created in a Computed Aided Design (CAD) package, but this process is time-consuming and expensive. For real-world objects an alternative is to employ a 3D acquisition system to create a virtual representation that can then be manipulated in software.

Due to its small working range, DFD would only suitable in this application for the capture and creation of small, virtual elements. However, DFD has been used for scene segmentation [48], and thus over large distances, where depth accuracy is less important.

3D TV

Britain was the first country in the world to have public television transmissions on a large scale, but the cost was very prohibitive. The technology drive during the Second World War helped to reduce the cost and it was estimated that over 20 million people watched the Coronation of Queen Elizabeth II on 2nd June 1953. About the same time colour televisions started to become a common feature in North American homes. Following the introduction of High-Definition TV (HDTV) the next big step is likely to be 3D-TV. Passive DFD has not been shown to work effectively over the large ranges required for 3D-TV, thus making stereo the obvious choice for the moment.

The increasing power of computer chips in digital television sets, the availability of 3D rendering hardware and developments in 3D display technology have led the BBC to speculate that 3D TV sets could be available in 2010 [49]. Different types of glasses have been designed for viewing 3D pictures, but each of them inconvenience the wearer and can cause discomfort when worn with existing corrective lenses. The concept of red-green (anaglyph) glasses was demonstrated by the Frenchman Joseph d'Almeida in 1858 [50]. The experience relies on the viewer being able to tolerate different colour images reaching each eye and only grey-scale or pseudo-colour images can be seen. In contrast, full-colour

3D images can be seen by a viewer wearing polarised glasses where two projectors (one for each eye) are employed with orthogonal polarises in front. LCD shutter glasses also produce colour 3D images and can be comfortably worn for long periods of time and they are becoming relatively cheap due to consumer games market for PCs [50].

Autostereoscopic display systems refers to technology that does not the require the viewer to wear specialist glasses. Most autostereoscopic displays are based on lenticular lenses. One of the main problems associated with integral imaging is the narrow viewing angle and this has been addressed [51].

Face Recognition

Anti-terrorism technology requires accurate ways of detecting known suspects who may have their information stored by the FBI, CIA or Interpol. Three-dimensional face recognition technology could help detect criminals at airports and other security check points [52]. As human heads could comfortably fall in the mean depth range used for DFD, it is possible that the technique could be used for this application.

Art Conservation

The conservation of sculptures is important in the arts world and whereas a painting can be photographed or scanned, a sculpture requires a 3D scanner; and for small sculptures, DFD could be employed to build the model. Once the 3D representation has been created it can be preserved indefinitely in electronic form and can be viewed easily from anywhere in world over the Internet.

Surgery

The visual inspection of the abdominal cavity using an endoscope, known as laparoscopy, was first performed in 1901 and during the 1970s the work of Kurt Semm in Germany showed that surgery was possible [53]. Advances in fibre-optics and video allowed the surgeon, and now their support team, to see clearer images inside the human body. Minimal invasive surgery leads to shorter recovery times for the patient, less scaring, less blood loss and a shorter hospital stay. The small movements of the surgeon have to be transferred through the specially designed instruments inserted through the small openings that may only be 1 or 2cm wide.

Robots have increased dexterity and steadiness over a human and at the control of a surgeon they have aided operations. At present the surgeon is likely to be in the next room, but remote surgery, known as *telesurgery*, has been investigated. The world's first

transatlantic telesurgical operation was carried out in September 2001 by doctors in New York on a patient in Strasbourg, France requiring the removal of a gall bladder [54]. The *Da Vinci Surgical System* is one of the very few robotic-assisted surgery tools on the market and it offers 3D visualisation to aid the surgeon. Surgery performed entirely by robots is a long way in the future, but clearly accurate 3D vision is going to be a major aspect of the research.

DFD is certainly a feasible method of obtaining 3-D images in laparoscopy as the working range is small. Further, the fact that it does not require two cameras separated by a sufficient baseline means that the opening in the patient may be smaller.

Planetary Exploration

In 1996 NASA launched the Mars Pathfinder mission with the first rover called Sojourner to explore the red planet. Viking I and II had been sent in 1975 and each had an orbiter and lander, the latter being incapable of motion after landing, but provided valuable biological, chemical, meteorological and geological data. The twenty minutes it takes the communications from Earth to reach Mars meant that a rover had to be autonomous to some extent. Sojourner had six wheels with which to roam the surface, it performed scientific experiments and benefited from nearly twenty years of innovations in image processing. In 2004, two rovers named Spirit and Opportunity returned with more advanced stereo imaging technology that enabled the rovers to plan routes, navigate autonomously and avoid obstacles.

As with virtual production, the problem of requiring depth estimates over long distances for producing large maps for terrain navigation may preclude the use of DFD, but for analysing smaller objects, such as rock samples, it may be a possible solution.

Disaster Exploration

Chernobyl Nuclear Power Plant was the site of the world's worst nuclear disaster in 1986 and a concrete and steel sarcophagus was hurriedly built over the Unit 4 reactor building to contain the radiation. The high radiation fields and structural instability precluded human examination when it appeared that it was starting to degrade, the fear being that contaminated dust would be released. A robot named Pioneer was used in 1999 to provide a 3D reconstruction of the interior using stereo videography that was based on Sojourner's software.

DFD may be able to provide 3D information of small components that must be inspected, but again, for producing models for large distances it is not suitable.

1.5 Research Objectives and Thesis Structure

1.5.1 Research Objectives

The aim of the research was to add knowledge to the field of depth-from-defocus concerning the improvement of depth accuracy. As DFD is based on defocus measurements, it is clear that accurate PSF measurements are required. The first part of the research was centred on obtaining accurate PSF measurements of a defocused camera system using the knife-edge based technique developed by Reichenbach *et al.* [55], Tzannes and Mooney [56] and Staunton [57].

As the Literature Review of Chapter 2 will show, no previous work on the use of colour images in DFD could be identified, where all three planes of two defocused images were used. The second section is concerned with the development of a depth-from-defocus algorithm that uses colour information, akin to acquisition of multispectral images using cones in the eye. Previous work was based on monochrome images, which is analogous to using the rods only.

The reason for investigating the use of colour images was firstly because it had not be looked at before. Secondly, it is known that colour images possess more information than monochrome images and the objective of the research became to see if the extra spectral information could be used effectively to achieve better depth accuracy with a DFD algorithm. The colour images were converted to monochrome in the pre-processing stage and then applied to an implementation of Ens and Lawrence's [58] [59] DFD algorithm, as shown diagrammatically in Figure 1.7.

Ens and Lawrence's elegant, matrix-based DFD algorithm was selected out of the alternatives presented in Chapter 2 and compared in Section 2.6. This was because it readily accepts experimental PSF data (which was the aim of the first part of the research), Ens and Lawrence showed that the spatial domain offered better depth localisation than the frequency domain for a given window size and their experimental results showed a good depth accuracy that put it 4th out of the 21 algorithms compared. Also, the relative simplicity of the algorithm was helpful in making the errors in the depth map easier to analyse and the implementation details were clearly available.

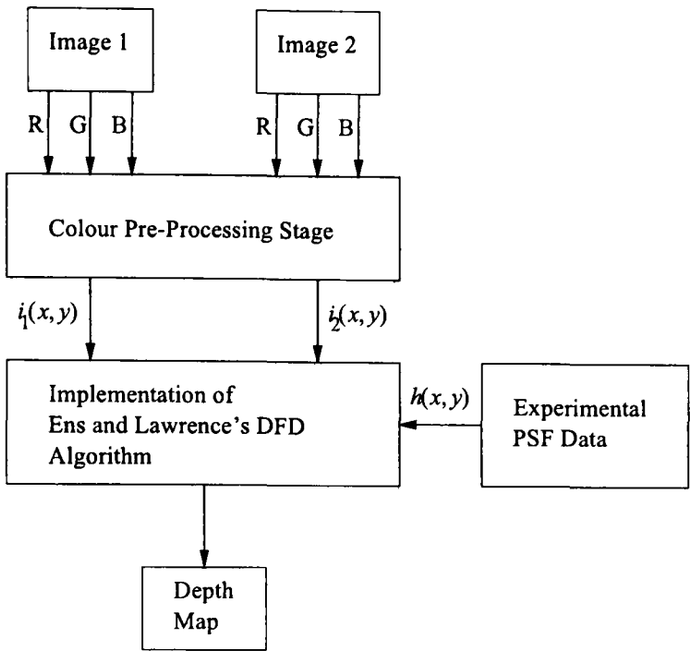


Figure 1.7: Flow diagram of the colour DFD algorithm

1.5.2 Thesis Structure

This thesis, entitled "Colour Depth-From-Defocus Incorporating Experimental Point Spread Function Measurements", is divided into two related sections, as discussed in the previous section.

A detailed and structured literature review of previous work on depth-from-defocus is presented in Chapter 2 from its generally agreed birth by Pentland [60] in 1982. Single and multiple image DFD algorithms are examined and the main problems with the concept are highlighted.

Theoretical models of the PSF have been derived ranging from the very simple pillbox model to the mathematically convenient 2D Gaussian. The accuracy of most depth-from-defocus algorithms is highly dependent on the precision to which the PSF is modelled and so the problem of accurately determining the PSF experimentally was considered. Chapter 3 begins with a survey of previous work to measure the PSF and presents derivations of the theoretical shape assuming geometrical and physical optics. Staunton's [57] method of determining a 2D PSF from a lightbox with a knife edge is discussed along with improvements that were made in different stages of the modelling.

Chapter 4 begins with results of the important linearity and noise experiments performed on the Basler A631fc colour camera. The experimental aspects of measuring the PSF are introduced and then results given for a 16mm video lens and a 24mm Sigma photographic lens over a range of distances.

A pre-processing stage that converts a colour image to monochrome is introduced in Chapter 5 with the purpose being to produce more accurate depth maps than just using a black-and-white camera. The different algorithms presented were designed to alleviate some of the problems with DFD discussed in the literature review.

Ens and Lawrence's algorithm [58] [59] based on a look-up table derived from experimentally determined PSFs was employed to test the pre-processing stage. Chapter 6 begins with a discussion of the implementation and then each of the different colour-to-monochrome pre-processing algorithms are presented and compared in turn.

In the final stages of the research it was discovered that the image normalisation, which was required when images with two different apertures were used, was sub-optimum. Chapter 7 presents solutions to the normalisation problem from theoretical and experimental perspectives.

Finally, Chapter 8 summarises the findings of the two parts of the research. Due to the limited time available it is inevitable that not all the possible research has been completed and the second section of Chapter 8 discusses future work.

Chapter 2

Literature Review on Depth-From-Defocus

2.1 Introduction

This chapter presents a comprehensive literature review of the work done on the $2\frac{1}{2}$D computer vision technique depth-from-defocus (DFD) using active and passive illumination. The active DFD algorithms require the use of a projector to ensure the scene has the required properties whereas passive systems use standard lighting techniques. The limited depth-of-field produced using a camera with a lens is presented in Section 2.2. DFD can be performed on a single image if strong assumptions can be made about the scene that the camera is imaging and Section 2.3 presents a review of the passive DFD algorithms using a single image. The brightness contribution of an image due to the scene can be separated from that due to defocus using two images. Section 2.4 examines the many passive DFD algorithms that have been designed to measure the defocus, and consequently the depth of the points in the scene, from two images. Active DFD methods are reviewed in Section 2.5 and finally a summary is presented in Section 2.6.

2.2 The Basic Premise of Depth-From-Defocus

Two monochrome images of a chessboard are presented in Figure 2.1 and Figure 2.2 and the only difference in the images is that the f-number of the camera was changed, resulting in different depths of fields. The exposure time was altered to compensate for brightness variations caused by the change of apertures. We can readily see the 3D nature of the chess pieces from the 2D image, firstly, because we hold the general assumption that the pawns are identical and since those on the left hand side are smaller then they must be further away (hence employing size perspective); secondly, defocus effects are acting as a depth cue.

Visual tests by Pentland [3] showed that scenes with greater amounts of defocus give the impression of a stronger sense of 3D structure. The reader may compare the images to see if they agree with Pentland's findings that Figure 2.2 gives a greater sense of three-dimensionality in the 2D images. Photographers frequently direct the viewer's attention to the subject of the image by defocus blurring the background [61].

Figure 2.1: Image of a chessboard taken with a small aperture (f/8)

Figure 2.2: Image of a chessboard taken with a large aperture (f/2.8)

Pentland noted that biological visual systems, such as the human eye, employ an optical system that produces defocused images, except for a small region in the centre of the retina, called the fovea. Importantly, it would be possible for the eye to have a smaller aperture (iris) and thus produce a much sharper image without incurring a significant loss in brightness [3]. It appears as though defocus effects improve the three-dimensional awareness of a biological system and provide more information than a pinhole image. Images taken with a small aperture have a large depth-of-field with the consequence being low depth discrimination [4].

Chromatic aberration of the lens in the human eye means that the focus position changes according to the wavelength of light and subsequently the eye produces images with different depth-of-fields in different spectral bands. Accommodation of the human eye was impaired when the chromatic aberration was reduced using an achromatising lens and monochromatic light, suggesting that the human brain uses chromatic aberration as a directional cue [62] [63]. Further, a sinusoidal change of the focal length of the lens with a frequency of approximately 2 Hz about the fixation point supplies the brain with extra focus information [64].

It is often stated in papers on DFD that the method eliminates the correspondence (matching) problem of stereo (e.g. [42]) and avoids occlusion problems. Schechner and Kiryati [65] have argued that it is not the case when considering geometrical optics and an analysis involving DFD, DFF, stereo and DFMB. Their analysis revealed that the chance of occlusion is higher with DFD and DFF compared to stereo and DFMB, however, the continuum of points caused by defocus blurring yields more information than disparity caused by a stereo system. Thus, DFD and DFF are more stable in the presence of occlusions.

The correspondence problem in stereo manifests itself as the existence of an ambiguity in the matching process and thus the triangulation, with the effect that there are multiple depth estimates for a given point. A spatial frequency analysis by Schechner and Kiryati [65] revealed that image regions could exist where the depth estimate using DFD is not unique, and thus it does not avoid the correspondence problem. The edge effect further highlights the fact that the matching problem exists. This is where the regions outside of the window blur sufficiently to contribute intensities inside the window, as shown in Figure 2.3, and the greater the defocus, the larger the effect.

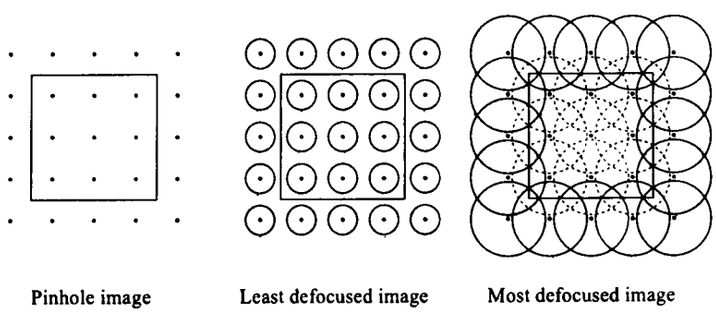


Figure 2.3: An illustration of the image overlap problem

As Ens and Lawrence [59] note, the concept of inferring distance of objects from defocus was reported by Helmholtz [66], but the first experimental work was performed by Pentland [60]. Twenty-four years have passed since Pentland's idea of depth-from-defocus was presented and in that time many different researchers have considered the problem and the next sections examine the progress that has been made to date. DFD

produces depth maps and thus the technique produces $2\frac{1}{2}$D images whereas volumetric techniques, such as CT and MRI, produce 3D images.

2.3 Depth-From-Defocus using a Single Image

Sharp intensity discontinuities in the scene, known as edges, provide useful information about depth because the brightness transition is primarily due to defocus. Pentland's first algorithm used a single defocused image of a scene and linear regression was employed to relate the Laplacian of the scene to a mathematical defocused step edge, from which the standard deviation σ of the Gaussian could be obtained [3]. As noted in Chapter 1, for a given lens there are two possible object distances that give the same σ, and the ambiguity is removed through knowledge of the set-up. The approach was implemented and shown to produce a depth map where edges were categorised according their distance: large; medium; or small depth values [3]. Although the approach is very simple and depth resolution poor, it could be employed for segmenting a scene. Lai *et al.* [67] used least-squares fitting to fit a linear step edge in depth convolved with a Gaussian PSF, which was a generalisation of Pentland's method and does not require differentiation.

Subbarao and Gurumoorthy [68] considered the problem of a defocused step edge in intensity and instead of using the standard deviation of the Gaussian, the square root of the second central moment of the PSF was employed. This approach essentially removes the restriction on the form of the PSF. The results showed that the spread parameter was linear with respect to inverse depth except near the focus position, where the difference was attributed to lens aberrations and the spatial and greylevel sampling. The approach was limited to isolated step edges as surrounding structure adversely affects the results and further it only worked for vertical edges due to the derivation.

Saadat and Fahimi [69] used the second derivative at the origin in the frequency domain as a measure of the bandwidth of the low-pass filter that defocused the scene and related it to depth. The algorithm was rewritten to use spatial domain integrations to reduce the effect of noise and it was assumed that the texture of the scene remains constant, allowing the single image approach to work.

Simon *et al.* [70] presented an algorithm to recover the depth of step edges using two images by relating the gradient of the edges to the standard deviation of a Gaussian PSF. The simulation results were good and fairly robust in the presence of noise, but the algorithm only worked for edges of multiples of 90 degrees due to the derivation, which is a serious deficiency. A more advanced approach to handle any angles was later derived and simulated on synthetic images [71].

The use of a defocus measurement based on a Sobel edge detector combined with motion detection information was used to yield foreground and background segmentation in a video sequence by Swain and Chen [48]. The motion detection meant that the background did not have to be static for correct segmentation.

Tsai and Lin [72] proposed a method where a Sobel edge detector is applied to the defocused image (so that the sharper the image, the greater the magnitude), the edge points are located and then a moment-preserving method is applied to give a binary image (with pixels representing high and low gradients) and then the proportion of the edge region in a small neighbourhood is found. From the neighbourhood measure and knowledge of the camera parameters the depth can be recovered, but the authors employ an Artificial Neural Network (ANN) to reduce the estimation error due to the effects of optical aberrations, vignetting and a PSF shape that differs from the assumed pillbox. The approach required a circular window with a radius of 35 pixels, which will clearly limit the spatial resolution of the depth map.

This review has shown that limited depth information can be obtained from a single defocused image, however, the problem with using a single image is that there is insufficient information to recover the defocus operator accurately. Image regions that have a smooth transition in brightness can be due to a defocused sharp edge or a focused soft edge [58], which could be the result of a gradient in the illumination for example.

2.4 Passive Depth-From-Defocus using Multiple Images

2.4.1 Introduction

Exploiting known properties of the scene, such as sharp transitions in depth, allows 3D information to be recovered as shown in the previous section. However, generally the scene is unknown and the contributions of the brightness as imaged with a pinhole camera cannot easily be separated from those due to defocus. Two images of the same scene taken with different camera parameters frequently allows the contribution of the scene to be factored out (assuming the scene is static), thus leaving the important defocus information. The exception occurs for textureless objects that appear identical in both images. Many passive depth estimation algorithms, including stereo and depth-from-focus, would also suffer from this problem, hence the use of active lighting, which is discussed in Section 2.5.

The accuracy of the estimates of the depth are dependent on the camera parameters chosen. Rajagopalan and Chaudhuri [73] analysed the error using the Cramer-Rao bound of the variance of the error assuming the images can be modelled using a 2D auto-regressive (AR) process and the blur was recovered using a maximum likelihood (ML) estimate. Interestingly, if the depth of the scene is not constant no single set of camera parameters may be optimum. Even assuming the depth of the scene is constant, in order to estimate the optimum camera parameters for the second image given the equation derived, an estimate of the blur in the first image, the AR parameters of the scene and the variance of the noise are required. If the required parameters were known, the authors showed in an example that the difference between choosing the best and worst camera settings resulted in a difference of 13% in depth error. As the accuracy is dependent on good estimates coupled with the other problems, it seems more reasonable to use a trial-and-error approach.

Two images taken with different f-numbers have the significant advantage that there is no magnification or image-to-image matching problem, unlike when changing the focal length or the distance between the image plane and the lens, although this can be remedied using a telecentric lens or image warping. As Watanabe and Nayar [74] point out, the two apertures must be very different in size, resulting in a darker image with the smaller aperture and having a lower signal-to-noise ratio (SNR). Further, the depth sensitivity is lower compared to changing the focal settings.

Many depth-from-defocus algorithms are highly mathematical, but it is important to keep sight of the problems encountered practically, with noise especially, as shown by Hwang *et al.* [75]. Their algorithm appeared correct mathematically but noise swamped the depth map to the extent that the results were showed as being either closer or further away than a given distance.

2.4.2 Pentland's Approaches to DFD

Pentland's second algorithm was formulated in the Fourier frequency domain, as it possesses the useful property that the spatial domain convolutions due to defocus become frequency domain multiplications. The PSF was assumed to be a 2D Gaussian with a standard deviation σ. The Fourier transform of a circular portion of the more focused image was divided by that of the more defocused image and then linear regression was used to estimate σ [3]. The images were multiplied by a Gaussian window to reduce the edge effect.

The main problem with Pentland's algorithms was that there is an assumption that one of the images is taken using a pinhole camera, which is generally unrealistic as it requires

a longer exposure time due to the tiny aperture and furthermore the effects of diffraction become more pronounced as the width of the aperture is decreased.

Pentland suggested that using more than two images allows for multiple estimates of the depth, thus affording the opportunity for error checking and possible averaging [3].

The problem with taking the Fourier transforms of both images is that it is expensive in terms of time and hardware and so the implementation used Parseval's Theorem. The images were convolved with a Laplacian band-pass filter to restrict the band of frequencies employed, the resulting pixel values were squared and then averaged using a Gaussian filter to give an estimate of the power in those frequencies. A look-up table was then employed to produce an estimate of the depth from the power images, which was found to have an error of 6% at 8 frames per second (fps). Instead of using a single band of frequencies a Laplacian pyramid could be used with linear regression to give a more accurate estimate, which was found to be 2.5% over a 1m^3 workspace [14]. The range estimates could not be considered dense though because only 64×64 measurements were taken. Pentland *et al.* [76] showed that post-processing the depth maps using regularisation (in this case using wavelets) could improve the final result.

Bove [7] modified Pentland's approach to use the pillbox PSF instead of the Gaussian and used a higher-order regression. A relaxation method was incorporated to deal with those image regions with insufficient high frequency texture so that the depth was consistent with neighbouring regions, as it was reported that those regions appear closer than is actually the case.

Taking the ratio of the Fourier transform of the images would lead to a single depth estimate and so the Short-Time Fourier Transform (STFT) was employed to ensure the depth was calculated for a local region. This requires a window to be used, which produces errors as the resolution is reduced in the frequency domain and the image overlap problem increases as the size of the window is reduced.

Xiong and Shafer [77] proposed an iterative method that seeks to blur one image using a Gaussian PSF to be the same as the other in the Fourier frequency domain, an approach they call *maximal resemblance estimation* that they claim eliminates the window effect. Although no quantitative results were given, smoother depth maps were produced using the iterative method compared to the direct method.

2.4.3 Subbarao's Approaches to DFD

Subbarao [6] generalised Pentland's work so that a pinhole image was no longer required and instead two images were taken with small changes in either the aperture size, focal length (which is achieved by changing the relative distances between lenses in a multiple component system) or the distance between the lens and image plane. The problem with employing small changes was that the images change very little and subsequently the depth map was less robust in the presence of noise, which is unavoidable in practice.

The algorithm was modified to allow for large changes simultaneously in any of the three variables using an average of the ratio of the power spectrum of the images in the Fourier frequency domain [4]. As described in Section 2.3 the measure of defocus proposed by Subbarao was the square-root of the second central moment of the PSF, thus allowing for any shape, as the Gaussian and pillbox models were not found to be an adequate [4].

Autofocus methods in cameras involve a search for the lens position that maximises a focus measure, but Subbarao and Wei [78] presented a technique that requires only two or three images to be taken. Furthermore, the image segment to be processed was summed along the rows giving an average that reduces noise as well as computational requirements. The Fourier transform was employed with a look-up table to produce the best focus position for the object and a RMS error of 6% was reported in terms of lens position. Instead of measuring the depth using the camera system, the corresponding focus position step was reported, which was linearly related to inverse distance.

Subbarao and Surya's Spatial Domain Convolution / Deconvolution transform (S-transform) was used to measure the defocus in two images taken with two different apertures, where the transform links the derivatives of a cubic polynomial approximation of a smoothed image to the moments of the PSF [79]. Each pixel in the region gives rise to a depth estimate and the mode depth was used [80]. A look-up table of the standard deviation of the PSF σ was generated as a function of depth in the calibration stage. Dense depth maps are produced by virtue of the local operations required and the RMS was 2.25 focus position steps out of 97.

In their review, Watanabe and Nayar [81] report that Subbarao and Surya's algorithm is suitable for large planar surfaces, but not for scenes where the depth variations are significant. The reason that Subbarao and Surya were able to quote a good depth accuracy was because their tests consisted of planar objects perpendicular to the optical axis and they did not report any results with step discontinuities. Surya and Subbarao state that DFD

could be used as a pre-processing stage for improving stereo vision [79] or depth-from-focus [82]. Yuan and Subbarao [83] presented a method that uses depth-from-focus and defocus techniques to generate a coarse depth map that is refined using colour stereo matching. Monochrome images were used for the depth-from-focus and defocus as the improvement was expected to be marginal using colour where the band with the highest contrast was employed.

An image with a larger depth-of-field than either of the images taken for DFD can be created by performing deconvolution using the known PSF shape following depth measurements. Subbarao *et al.* [84] showed that the S-transform and inverse Abel transform successfully performed this task.

The S-transform created by Subbarao and Surya [79] was a specific case of a more general algorithm developed by Ziou and Deschênes [85] [86] that employed a local image decomposition technique based on Hermite polynomials. The best fit polynomial representation of the most blurred image was a function of the partial derivatives of the less blurred image and the difference in the standard deviations of the Gaussian PSF that defocused the images. The difference in the standard deviations was linked to the camera parameters to find the depth of a point in the scene. Ziou and Deschênes claimed that their approach yielded a smoother, denser and more accurate depth map. For a planar object between 115 and 125cm away the RMS error was reported to be 2.21%.

An interesting point that was not made by Subbarao, but also applies to Ziou's work too, is that their algorithms cannot determine the depth when a step edge is present or constant intensity junctions, such as L, T, V, X and Y shapes, as the Laplacian of the Gaussian is zero [86].

The concept of modelling the images using a polynomial was continued by Rayala *et al.* [87] who formulated the DFD problem as that of identifying the parameters in a system model with the least defocused images forming the input and most defocused image forming the output. By approximating the images, the second derivatives could be easily found, from which the spread of the PSF could be recovered.

Deschênes, Ziou and Fuchs [88] modified their algorithm based on Hermite polynomials to allow for spatial shifts between the defocused images with the motivation being that it is very difficult to perfectly align the cameras and ensure no vibration. The RMS error for the plane decreased from 2.21% to 1.68% with the modified algorithm.

As discussed above, Subbarao [6] considered DFD by changing the camera parameters by infinitesimal amounts and found that the errors were significant. Farid and Simoncelli [89] proposed a similar, derivative-based approach, but added attenuation filters with variable-opacity in the optical path to take two images from the same camera position.

The PSF then becomes a scaled and dilated version of the mask (created using a programmable liquid crystal spatial light modulator). The depth was assumed to be constant within a window and it was linked mathematically to the images created with the masks. The standard deviation of the error was 0.06cm and 0.16cm for planes 11cm and 17cm from the camera respectively.

2.4.4 Ens and Lawrence's Approach

Spatial domain convolution becomes multiplication in the Fourier frequency domain and this has the effect of simplifying the extraction of the defocus operator as shown by Pentland [3] and this approach is called *inverse filtering*. Ens and Lawrence [58] showed that the cost of the simplification is that in order to drive the error in the shape of the PSF down to 1%, the window had to be an order of magnitude larger than the spatial extent of the PSF. This is an undesirable consequence of using the frequency domain because for good depth map resolution the windowed region needs to be as small as possible. In order to remedy this problem, Ens and Lawrence reformulated the DFD problem in the spatial domain using matrices. In the noise-free case, the window size needs to be no larger than the extent of the widest PSF.

Ens and Lawrence [58] [59] presented three spatial domain approaches of varying complexity. In the first and simplest case where there was no noise present, which can only occur in simulation, direct deconvolution using matrices was employed. In the second case a matrix-based regularised form was found where the PSFs were constrained to have a particular shape, for example a Gaussian. The third approach used an iterative approach that searches for the optimum function, known as the convolution ratio, from a pre-computed look-up table from which the depth can be derived.

The iterative matrix approach using the look-up table was tested on an inclined plane between 0.8 and 0.95m from the camera and the RMS error was reported to be 1.3% over that range, whereas the regularised form had an error of 6.8% [58]. As Horii [5] noted, the matrix-based approach had a high computational cost, which he did not think was worth paying because the accuracy was very dependent on the signal-to-noise ratio (SNR).

2.4.5 Watanabe and Nayar's Approach

Depth-from-defocus requires accurate measurement of the defocus between images and this generally requires a large filter bank to sample the Fourier frequency space sufficiently. Watanabe and Nayar [81] proposed the use of a set of broadband filters (the rational operators based on the equifocal assumption) with a support of 7×7 to produce accurate and dense depth maps that are invariant the scene texture. A single broadband filter (as used by Pentland for example) cannot produce accurate results because there are two unknowns: the response of the defocusing low-pass filter; and the texture of the scene. The defocused images are pre-processed to remove the DC and very high frequency components. The pillbox PSF model was employed to model the relative blur in the frequency domain. The confidence in the depth is derived from the operators, which in turn allows for refinement at a post-processing stage. The reported depth accuracy was between 0.5 and 1.2% of the distance from the object to the camera. Although a real-time implementation was not presented the algorithm is clearly efficient and could be built on standard hardware.

2.4.6 Xiong and Shafer's Approach

The spatially-varying nature of the PSF is more easily analysed by windowing the image to obtain a small region for processing and then assuming that the depth is constant within the window. Taking a small region of the image introduces windowing effects and further, it cannot necessarily be assumed that the depth is constant. Xiong and Shafer [90] introduced two new sets of filters, called *moment* and *hypergeometric* filters, that possess recursive properties that help to eliminate the windowing problem and not just remove foreshortening (which was the term they used for the non-stationary nature) effects, but also measure the degree. In addition to the properties mentioned, hypergeometric filters produce a complete and non-redundant decomposition of the signal (or image).

The moment filters were applied to the problem of depth-from-defocus and for a sloping plane the RMS error was found to be 27 times better using the moment filters incorporating the space-variant PSF compared to Subbarao's algorithm proposed in [4]. The hypergeometric filters were not tested in DFD, but improvements would be expected with those too. The main drawback with the approach is that a lot of computational time and memory was required and Xiong and Shafer state that parallel computers are necessary really. Essentially it is not amenable to a hardware implementation as it requires many filters [91] and the accuracy is determined by how well the optical system was modelled.

2.4.7 Rajagopalan and Chaudhuri's Approach

Many of the DFD algorithms are based on the assumption that the depth is constant in a small window, thus assuming a space-invariant blurring function. In fact this is an approximation of the real (general) case where the depth is changing and the blurring is space-varying. Two space-variant approaches were proposed by Rajagopalan and Chaudhuri [92]. The first algorithm, known as Block Shift-Variant (BSV) is based on the assumption that the blur is constant within a subimage, but variant over adjacent subimages. This is essentially the same assumption as Pentland [3], Subbarao [68] and Ens and Lawrence [59] except that the interaction between blocks was considered.

The second set of algorithms were based on space-frequency representations using the Wigner distribution and the complex spectrogram [92]. The results were better than the BSV approach, but all three algorithms produced poorer results at depth discontinuities.

Rajagopalan and Chaudhuri [93] modelled both the images and the depth map using Markov Random Fields (MRFs) and used simulated annealing to find the Maximum A Posteriori (MAP) estimates. A focused image and a depth map were produced by the algorithm. Subbarao's algorithm [4] was used as a benchmark in a real scene and found to produce an error of 6%. Their algorithm performed better, giving a depth error of 4%, but with some more information about the scene or the depth field the authors believe the results could be improved further. As Favaro and Soatto [94] note, the use of MRFs is effective, but suffers from a high computational cost.

The sampling of the image capturing device, such as a CCD, imposes a restriction on the spatial resolution of the defocused images and consequently the depth map recovered using DFD. Rajan *et al.* [95] proposed a super-resolution approach where multiple images are taken with sub-pixel camera movements between each image. The depth map and the brightness images were modelled using MRFs and a cost function was minimised to calculate the parameters.

2.4.8 Favaro and Soatto's Approach

Favaro and Soatto [96] developed an iterative DFD algorithm that estimates both the shape of the scene (i.e. the scene's geometry) and its reflectance properties simultaneously. The formulation is based on creating a model of the scene and then a measure of the error between the actual image and the model is minimised. Commonly employed measures are the squared-distance (corresponding to a least-squares problem) and total variation, which is based on the integral of the absolute value of the error. Based on the work of Csiszár [97] the measure chosen was the information-divergence.

Jin and Favaro [8] improved the original algorithm to allow for a space-variant kernel (PSF) and the scene radiance and geometry was obtained through solving partial differential equations (PDEs). An iterative procedure generates the global image step-by-step and the regularisation process ensures scene smoothness. Although no quantitative results were presented, a scene consisting of figurines with a continually changing depth was recovered very well.

Favaro and Soatto [98] proposed a matrix-based formulation of DFD where the process of defocus was learnt, as opposed to being modelled by the interactions of a PSF and a representation of the scene. A training set of images of a defocused plane at a particular depth was created where the radiance of the plane was changed. Singular Value Decomposition (SVD) was then employed to find similarities between the training images and an orthogonal projector operator was created so that images generated by the same shape with any radiance belong to the null space. The operators for planes of different depths are also found. With the lookup table of operators complete, unknown scenes can be processed. The depth of a point in the scene was estimated by searching for the operator that lead to the minimum residual. Interpolation can be applied to increase the number of different depths. A plane was moved from 0.52m to 0.82m in 51 increments and the depth computed giving an RMS of 3.78mm.

If the form of the PSF was known then the learning process was not required and functional SVD was employed to compute the required orthogonal projectors [94]. The advantages of the approaches are that a small window is required (7×7 or 9×9), the algorithm is robust to noise and a real-time implementation is feasible. Further, the shape of the PSF is not required for the learning approach.

2.4.9 Voxel Approach

A *voxel* is a 3D counterpart to a pixel (picture element) and Prasad and Mammone [99] formulated the DFD problem as turning voxels on and off to create a depth map where simulated annealing was employed to solve the constrained optimisation problem posed.

2.4.10 Entropy-Loss Formulation

The effect of defocusing is to reduce the high frequency content of an image, a property that is exploited in depth-from-focus algorithms, but from an information-theoretic point of view, defocusing decreases entropy, which corresponds to an increase in the statistical correlation of neighbouring image points. Bove [100] used the entropy loss as a measure of defocus and related it to depth. A wallpapered plane was used to test the accuracy of the algorithm and it was moved between 1.3 and 2.0m from the camera. The RMS error was 2.2% in terms of measured distance from the camera and 5.2% when the expected range was considered and in comparison Bove's higher order regression approach [7] discussed in Section 2.5.1 produced RMS errors of 2.5% and 5.8%.

2.4.11 Dynamic-Referencing Approach

Horii [5] used division in the Fourier frequency domain and designed an implementation based on using Parseval's theorem and a Laplacian filter to extract the power in a restricted range of frequencies, like Pentland, but proposed a solution to remove the texture dependency. One of the images is dummy blurred with a Gaussian filter to determine a required constant in a method called *dynamic-referencing*.

2.4.12 Artificial Intelligence Approaches to DFD

Fuzzy logic is an artificial intelligence method that seeks to use qualitative, instead of quantitative, data. Swain *et al.* [101] argued that fuzzy logic can be successfully applied to reduce the problems of noise, lens aberrations, varying lighting conditions, computational error and imprecise data due to lower resolution. Two variables were employed that begin as quantitative measures and then fuzzified using experimentally determined fuzzy membership functions. The first measure is focus quality based on the Tenengrad operator and the second is the focal error, which is measured using the Laplacian operator on each of the two images. The output of the fuzzy logic is a correction factor to apply to the depth map. The error was reported to be 1.5% over a range from 7 to 11 feet.

2.4.13 Wavelet-Based Approaches to DFD

All of the frequency domain approaches to DFD employ the Short-Time Fourier Transform (STFT) in an attempt to localise both the spatial and frequency information. The spectrum of the image segment is convolved with the spectrum of the window due to the truncation, thus increasing the uncertainty. Wavelet analysis seeks to optimise the window size so that large window sizes are employed to find precise low frequency information and similarly small windows for high frequencies. Kim *et al.* [102] related the spread of the Gaussian PSF to the wavelet coefficients and demonstrated better results than using either a frequency domain or a spatial domain approach. In particular, the very difficult shape of a cone with its centre lying along the optical axis was recovered fairly well using the wavelet approach and badly using the other algorithms implemented. Hor *et al.* [103] also showed that wavelet approaches perform particularly well on space-variant problems.

2.4.14 Depth-From-Defocus Using Colour Cameras

Garcia *et al.* [62] used the inherent chromatic aberration of a lens and a colour CCD to capture the image to produce an RGB image where each colour plane has a different focal length. A measure of the spread of an edge was found for each colour plane from which it can be determined if the object is in front or behind the focal point of the camera and secondly, depth can be calculated through a mathematical relationship.

The PSF is determined by the shape of the aperture and Farid and Simoncelli [89] employed attenuation masks to allow the range to be recovered through differentiation. In a similar approach, Hiura and Matsuyama [104] modified the aperture to create a *coded aperture* that contained multiple holes, as they state that the blurring should be designed to yield accurate and reliable depth maps, as opposed to accurately modelling the given blurring. The coded apertures ensured that the important high frequency information was retained and they are placed to form a telecentric system to eliminate magnification effects. They also used a 3 CCD colour camera to capture three images with different focal lengths (a system they called the multi-focus camera). Two holes (instead of the usual single aperture) are employed and the depth is recovered through division in the Fourier frequency domain, but a look-up table is employed to determine the object distance. The RMS error was reported to be about 5% using the multifocus camera alone and no quantitative results were given incorporating the coded aperture. The redundant information from the three colour planes was then used to find the focused image of the scene.

Particle Image Velocimetry (PIV) is important for research in studying unsteady flows and the depth of particles has been found using stereoscopic systems, but the correspondence problem is particularly troublesome. Murata and Kawamura [105] proposed a DFD-based system to overcome the problem with the unusual addition of a colour CCD where the focal lengths are different for the red and green planes using colour filters and movable mirrors. The captured images were low-pass filtered to reduce the effect of noise and the relative defocus between the two colour planes was used a measure of the depth of a particle.

2.5 Active DFD Methods

2.5.1 Introduction

The passive methods discussed in the previous section are reliant on the scene possessing sufficient texture for DFD to work and where the requirement is not met the depth measurements are likely to be highly erroneous. The problem can be overcome using a structured light source to impose a texture on the scene using a data or slide projector, for example. The depth measurement results are affected by the choice of the specific form of the structured lighting and is thus subject to research.

The image overlap problem can be eliminated by projecting a pattern onto the scene that has bright areas that are examined for defocus surrounded by dark guard bands that ensure there is no contribution due to those area [4]. This will reduce the depth map spatial resolution, so a few such projections could be employed where the pattern is shifted each time and the resulting scene imaged at each step.

2.5.2 Pentland's Approach

The first active DFD algorithm was presented by Pentland *et al.* [76] and it employed a slide projector to produce parallel vertical lines on a scene focused at a set distance. The camera had a small aperture so that the pattern defocused with depth and is not complicated by defocus by the camera too. The moment of inertia of a small region centred on a defocused line imaged by the camera was related to the standard deviation σ of the Gaussian PSF that was assumed and hence depth could be extracted. An RMS error of 0.5% was reported and the use of a stroboscopic light source was discussed for capturing images of a rolling sphere.

2.5.3 Watanabe and Nayar's Approach

Nayar *et al.* [12] [91] argued that a precise model of the optical system was required for accurate DFD work and that it was necessary to provide active illumination with high spatial frequencies. The projection pattern was designed using a model of the system, which took into account sampling, diffraction, defocus and the focus measure employed (with a 3×3 kernel). Two optimum patterns were returned: one of which was composed of black-and-white squares (a checkerboard pattern) that are the same size as the pixels and with no phase shift; the second was composed of squares twice as large with a phase shift of half the sensor spacing. A look-up table relates the ratio of the convolution of the defocused images with the 3×3 kernel to depth. Real-time depth maps (30Hz) of a 512×480 pixel image were generated using hardware with an RMS error of 0.3% and the use of a telecentric aperture avoided magnification effects

2.5.4 Ghita and Whelan's Approach

The optimum patterns proposed by Nayar *et al.* [12] are difficult to make and Ghita and Whelan found that using image interpolation reduced some of the problems caused by a non-optimum pattern [106] [107]. The range sensor they constructed produces 256×256 depth maps at 10 frames per second using a Laplacian-based approach as proposed by Subbarao *et al.* [82], with a normalisation proposed by Nayar *et al.* [12] and with a projected pattern consisting of horizontal lines, like Pentland *et al.* [76]. Ghita and Whelan investigated the depth estimation performance using 4- and 8-neighbourhood Laplacian and 3×3 and 7×7 kernel rational operators [106]. They report that the 7×7 rational operator performed the best, but lacked linearity. Discontinuities in the depth were not well recovered using the 4-neighbourhood Laplacian and 3×3 rational operator, but the depth was more linear. The reported error was 3.4% in terms of the distance between the sensor and the scene.

2.5.5 Ma and Staunton's Approach

Ma and Staunton [108] developed an Artificial Neural Network (ANN) based approach that combined image segmentation and depth estimation. Multiresolution image segmentation was used to isolate object regions from the background. Lower resolution information, which was found to be strongly correlated to depth, was fed into a three-layer ANN as feature vectors and then processed to give a depth estimate. Although the approach required active illumination, it was simulated by gluing printed texture to the objects.

2.6 Conclusion

This literature review has shown that absolute depth information can be obtained using the various different formulations of DFD. The single image approach is very simple and it has been successfully used to aid segmentation of video sequences. However, out of six papers reviewed, only one had quantitative results and that was only from simulation results.

The passive and active DFD algorithms using two images allow for dense depth maps and their accuracies are presented in Tables 2.1 and 2.2 for comparison. The errors have been expressed as the RMS error divided by the range employed as this allows for a more direct comparison that the MSE alone. Where the closest and furthest points from the camera have been reported, it is reproduced in the table below. However, if only the range is given then it is given as a single number.

Of the passive techniques reviewed, the algorithms developed by Watanabe and Nayar, Xiong and Shafer, and Kim *et al.* have the same accuracy of 0.5%. Two of the three active methods with quantitative error results have identical accuracy to the best passive results, but Nayar *et al.* [12] have produced the only real-time range sensor (running at 30 frames per second) based on DFD that has a dense depth map. Pentland's simpler technique using projected lines claims the same accuracy, however, the depth map is not as dense and it was not implemented for real-time operation.

The mean working range of the passive algorithms reviewed was found to be 1.1m to 1.8m. The largest distance an algorithm was quantitatively tested on was Surya and Subbarao's STMAP algorithm [79] at 5m, but the accuracy was poor at 20%.

Table 2.1. Accuracy for passive DFD techniques

Author	Algorithm Name	Working range / m	Accuracy / %
Watanabe & Nayar [81]	Rational filters for passive DFD	0.540 - 0.840	0.5 - 1.2
Xiong & Shafer [77]	Maximal resemblance estimation	2.5	0.5
Kim *et al.* [102]	Wavelet analysis approach	1.50 - 1.80	0.5
Ens & Lawrence [59]	Iterative matrix approach	0.80 - 0.95	1.3
Swain *et al.* [101]	Fuzzy logic-based	2.1 - 3.4	1.5
Surya & Subbarao [79]	STMAP	0.6 - 5.0	1.6 - 20
Deschênes *et al.* [88]	Incorporation of spatial shifts	1.15 - 1.25	1.68
Rajan *et al.* [95]	SR from defocus blur	0.73 - 0.97	2
Bove [100]	Entropy-based	1.3 - 2.0	2.2
Farid & Simoncelli [89]	Optical differentiation	0.11 - 0.17	2.2 - 2.4
Ziou & Deschênes [86]	Hermite polynomial-based	1.15 - 1.25	2.21
Pentland *et al.* [14] [76]	Multi-scale	1.40 - 2.70	2.5
Bove [7] [100]	Higher-order regression (multi-scale)	1.3 - 2.0	2.5
Rajagopalan & Chaudhuri [93]	MRF with MAP	0.70 - 1.25	4
Rajagopalan & Chaudhuri [92]	Complex Spectrogram method	0.70 - 1.25	4.7 - 8.8
Rajagopalan & Chaudhuri [92]	Pseudo-Wigner Distribution	0.70 - 1.25	4.9 - 9.2
Hiura *et al.* [104]	Multi-focus camera	1.50 - 4.10	5
Rajagopalan & Chaudhuri	Block-Shift Variant method	0.70 - 1.25	5.4 - 10.6
Horii [5]	Dynamic referencing	1.5 - 2.5	5.8 - 8.4
Pentland *et al.* [14]	Single-scale	1.00	6
Ens & Lawrence [59]	Constrained inverse filtering	0.80 - 0.95	6.8

Table 2.2. Active DFD techniques comparison

Author(s)	Technique name	Working range / m	fps	Depth map size / pixels	Accuracy / %
Nayar *et al.* [12] [91]	Real-time focus range sensor	0.30	30	512×480	0.5
Pentland *et al.* [76]	Projected lines	1.40 - 2.70	-	64×64	0.5
Ghita & Whelan [106]	Real-time depth sensor	0.09	10	256×256	3.4

The algorithms can be catagorised into those that use the spatial, Fourier frequency and wavelet domains. A primary difference between the algorithms is that some assume a space-invariant PSF within a window (known as the *equifocal* approximation) and others assume a space-variant blurring kernel, which removes the depth restriction. The image overlap problem can be reduced by windowing with a centre-weighted mask (e.g. a Gaussian) to reduce the effect of the pixels at the edge [42]. When a space-variant PSF is employed, the radiance of the scene and its geometry (i.e. depth) must be computed, whereas in the equifocal case the geometry alone can be recovered [94]. Watanabe and Nayar [81] state that the reason passive DFD methods are computationally expensive is that the frequency characteristics of the scene are largely unpredictable. By virtue of the projected image, the frequency content of the scene with an active DFD system is known very well.

The advantages of DFD are that the accuracy is comparable to that using methods based on stereo disparity and motion parallax; and DFD is more stable in the presence of occlusions than stereo. As with stereo, matching (or correspondence) problems exist in DFD and in this case it is due to edge bleeding caused by the spread resulting from defocus effects.

The main disadvantages of DFD are that the shape of the PSF must be accurately known, windowing effects due to a space-variant PSF lead to inaccuracies and further sufficient texture is required on the scene, although this can be alleviated using structured lighting. The sensitivity to error is also dependent on the camera parameters used, aberrations present in the optical system and the spatial and grey-level resolution of the cameras [4].

Subbarao suggested a robot vision system that employs cameras with short focal length lenses for objects that are close and longer focal lengths for those objects further away [6] as the usable and accurate depth range is highly dependent on the camera parameters. For

example, a scene with a large depth of field will have low defocus discrimination [42]. Research involving the integration of different three-dimensional techniques is showing improvements, for example depth-from-defocus, focus and stereo [83] and focus, vergence and stereo [109].

In 1992 Horri [5] stated that the main purpose of DFD was to be to create a rough depth map for use in vergence, auto-focusing or stereopsis range finding algorithms, but he did not foresee the improvements that would be made in the field.

Chapter 3

The Theory of the Measurement of the Point Spread Function of a Defocused Imaging System

3.1 Introduction

Depth-from-defocus (DFD) algorithms rely on the limited depth-of-field produced using a real imaging system. The limited depth-of-field is caused by defocus blurring that is the result of space-varying convolution with a low-pass filter, known as the Point Spread Function (PSF). The PSF is dependent on the camera parameters and the depth of the point in the scene. Hence, knowledge of the PSF allows depth-from-defocus algorithms to determine the depth of a point in an image. Accurate measurement of the PSF is required for precise depth estimates [81] and these must be determined experimentally as no theoretical model can adequately take into account all the factors present in an optical system. This chapter focuses on the determination of the PSF for the Basler A631fc colour camera with a 16mm video and a 24mm Sigma photographic lens that was used in the subsequent DFD experiments. However, the methods also apply to any imaging system composed of a focusing optics (e.g. a lens) and a sensing array (such as a CCD).

An overview of measurement techniques for the PSF and its Fourier transform counterpart, the Optical Transfer Function (OTF), are described in Section 3.2 and then the theoretical PSFs assuming both geometrical optics and diffraction-limited optics are developed in Section 3.3. The measurement of the PSF can be achieved from differentiating a step response, known as the Edge Spread Function (ESF), and Section 3.4 presents models of the ESF for a given PSF. Finally, Section 3.5 summarises the findings and in Chapter 4 the results of applying the theory are shown.

3.2 Literature Review

3.2.1 Introduction

Pixel arrays, such as CCDs, are devices composed of many photosites (active areas) and their associated transfer electronics for readout of the charge and timing. The solid-state active area of a CCD that converts incident photons to electron-hole pairs are typically square, rectangular or L-shaped. The ratio of the active area to the total area is referred to as the fill-factor and front illuminated arrays have fill-factors less than 100 percent [110] as communications and other sub-systems take up a finite area.

A photon is emitted by a source, for example an incandescent bulb or a distant star, and it is reflected and refracted by objects before impinging on a CCD having been refracted by a lens usually. If the photon strikes the active-area of the CCD it generates an electron-hole pair and the resulting free electron is known as a photoelectron. The accumulated photoelectrons are stored in a potential well. The charge packets are moved from site-to-site during read out and in modern CCDs the Charge Transfer Efficiency (CTE) is very close to one hundred percent [111]. Each packet is amplified and then an analogue-to-digital converter (ADC) produces a digital output signal that is a function of the number of photons that struck a given site.

The Pixel Response Function (PRF) is defined as the output of a pixel as a function of the spatial position of a point source of light and thus it gives a measure of the sensitivity of a pixel as well as the crosstalk between neighbouring pixels. An ideal PRF has a uniform sensitivity within the boundaries of the pixel and zero outside so that there is no crosstalk, however, it has been shown that the sensitivity is a function of position within a single pixel and further it is a function of wavelength too. Figure 3.1 shows an example of an experimentally determined PRF for a $9 \times 9\,\mu m$ pixel by Kavaldjiev and Ninkov [112] where the sensitivity is shown in standardised units in the range [0, 1].

The variations between pixels are primarily due to transmittance non-uniformity, variations in the quantum efficiency and diffusion spreading of the photogenerated minority carriers [112]. Due to the non-uniformity in the PRF the response due to a point source is space-varying and it can be quantified using a shift-error, but the effect is reduced in a defocused system [112].

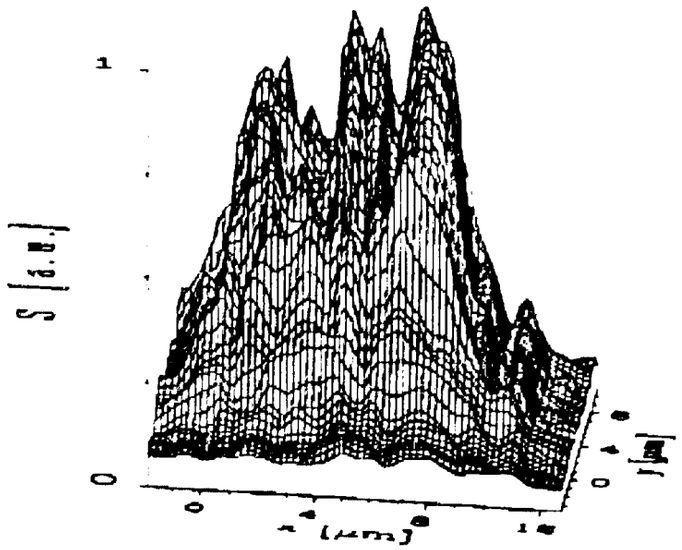


Figure 3.1: The experimentally derived PRF for a $9 \times 9\,\mu$m pixel in the Kodak KAF 4200 CCD at $\lambda = 633$ nm (from [112])

Measurement of the PRF is time consuming as potentially each pixel must be characterised individually due to non-uniformities of materials and the manufacturing processes. The measurement must be done for many wavelengths and requires high precision equipment as a point source must be moved at the sub-micron level.

The Point Spread Function (PSF) of an image acquisition system takes into account the response of the pixels as well as the optical and electronic elements in the system. Although the PSF can be found by convolving a model of the PRF with the PSF of the optics [113] and factoring in the response of the electronic systems, often the average PSF is measured using techniques discussed in the next section. The Fourier transform of the PSF is called the Optical Transfer Function (OTF), which is generally a complex function. The magnitude of the OTF is called the Modulation Transfer Function (MTF) and the phase component is called the Phase Transfer Function (PTF), but for a centred optical system the phase is often assumed to be zero.

The MTF gives a measure of the quality of the imaging system as it is a measure of the spatial resolution. It can determined for a complete system composed of a Focal Plane Array (FPA), such as a CCD, and the optics using techniques outlined below, but some of the techniques can be used to measure the MTF of the FPA alone, i.e. without any optics.

3.2.2 PSF and MTF Measurement Techniques

An image of a sinusoidal grating is a classic MTF measurement technique where the input illumination $I(x)$ as a function of spatial displacement x is given by

$$I(x) = \frac{1 + \cos(x)}{2} \tag{3.1}$$

and the MTF for a given spatial frequency ξ is given by

$$\mathrm{MTF}(\xi) = \frac{I_{\max} - I_{\min}}{I_{\max} + I_{\min}} \tag{3.2}$$

where $I_{\max}$ and $I_{\min}$ are the minimum and maximum pixel intensities. A focal plane array collects incident light over a given area and the sampling aperture effect of a discrete sensor means that the FPA must be moved relative to the target to give minimum and maximum MTF curves [114]. A square-wave target, as opposed to a sinusoidal grating, can be employed and the measured quantity is the Contrast Transfer Function (CTF) [24].

A problem with using printed test patterns is that the spatial frequency is fixed and so multiple patterns must be produced to span the required range. Sinusoidal interference patterns resulting from the interaction of two laser beams can be generated and were previously used for film cameras before being migrated over to CCD-based systems by Marchywka and Socker [115]. The creation of a continuously-varying sinusoidal interference pattern for measuring the MTF of an FPA alone has been demonstrated [116]. The main problem with using lasers to create the image is that the MTF is only determined for essentially one wavelength of light and it is known that the MTF is wavelength-dependent.

The spatial domain image $f(x, y)$ formed on a FPA is given by the convolution of the scene intensity $s(x, y)$ and the PSF of the optical system $h(x, y)$, thus

$$f(x, y) = s(x, y) * h(x, y) \tag{3.3}$$

where $*$ denotes linear convolution and transforming to the Fourier frequency domain gives

$$F(\omega, \nu) = S(\omega, \nu) H(\omega, \nu) \tag{3.4}$$

where $f(x, y) \overset{\mathrm{FT}}{\longleftrightarrow} F(\omega, \nu)$, $s(x, y) \overset{\mathrm{FT}}{\longleftrightarrow} S(\omega, \nu)$ and $h(x, y) \overset{\mathrm{FT}}{\longleftrightarrow} H(\omega, \nu)$ and $H(\omega, \nu)$ is the OTF of the system. If the spectrum of the scene is known then the OTF can be recovered and further if the scene is white noise then $S(\omega, \nu) = 1$ and the Fourier transform of the output $f(x, y)$ is the OTF. The laser speckle effect can be used to provide the required spectrum of the scene and the measurement technique was developed by Boreman and Dereniak [117] where the speckle is a result of interference of the coherent, monochromatic laser beam. The MTF produced is an average for the entire FPA and no optics are required. An integrating sphere and an aperture are used to control the frequency content [118]. The output port of the integrating sphere has uniform irradiance and the phase is randomly distributed [119], thus producing an average of the MTF for a space-varying system. The problem with the technique is that lasers with a power of around 200mW are required. Ducharme and Boreman [120] reported that speckle patterns can be created on a hologram and then a low power laser can be used to illuminate the hologram for MTF

measurements, where the reduced power requirements means that a wider range of wavelengths can be tested.

The use of lasers for speckle-based MTF measurements means only one wavelength is tested in a single trial. Daniels *et al.* [121] created random patterns on a computer and printed the random transparency targets that can be illuminated using any suitable, spatially uniform light source to measure the MTF of an FPA and optical system, allowing an average PSF to be measured for many wavelengths.

An ideal test for the measurement of the PSF is a point source that has infinite intensity and infinitesimal spread, often denoted a delta function $\delta(x)$, which is given by

$$\delta(x) = \begin{cases} 0 & x \neq 0 \\ \infty & x = 0 \end{cases} \tag{3.5}$$

and it is physically unrealisable. The running integral of a delta function denoted

$$u(x) = \int_{-\infty}^{x} \delta(x)\, dx \tag{3.6}$$

is a Heaviside step function given by

$$u(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \tag{3.7}$$

and this function is a step in intensity, which can be produced experimentally, using a lightbox for example. The response of the system to a step function is known as the Edge Spread Function (ESF) and differentiating the response gives the PSF. A sharp transition is necessary and so PSF measurement methods based on using this step function are often referred to as knife-edge techniques. Under-sampling effects cause errors in the OTF estimate due to the space-varying response and aliasing. Reichenbach *et al.* [55] solved the problem by using many ESF profiles to create a super-resolution image of a 1D edge. Tzannes and Mooney fitted a sum of three Fermi-Dirac functions to the edge to reduce the noise during differentiation [56]. Staunton extended the technique to measure the ESF for many different angles to produce a 2D MTF [57].

The concept of producing super-resolution ESFs by nearest neighbour interpolation is illustrated in Figure 3.2. In the diagram four ESFs are shown as lines with equally spaced data points. The data points do not correspond to those of the sampling grid and so the intensities of each data point are determined using the nearest sampling point, thus implementing nearest neighbour interpolation. The over-sampling means that a super-resolution ESF can be produced.

One of the problems with using the knife-edge techniques is that the two-dimensional PSF or MTF must be built up in a single 1D profile at a time. Reimann *et al.* [122] showed that the 2D MTF could be recovered in one step by imaging a precise circle and

using Wiener filtering to recover the MTF. The main problem with their technique is that the space-varying nature of the MTF is neglected.

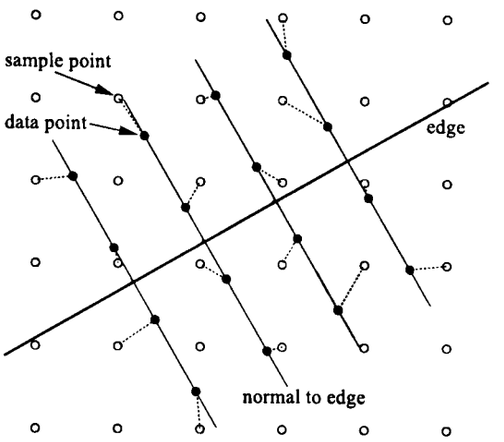


Figure 3.2: ESFs created from sampled data using nearest neighbour interpolation

Subbarao [68] used the square root of the second central moment of the line spread function, denoted σ, imaged by a camera viewing a black-and-white cardboard step to determine a spread parameter. Only nine different focus positions were used, but the results clearly showed that σ was directly proportional to the inverse depth.

3.3 Theoretical Point Spread Functions

3.3.1 Introduction

Photons have a wave-particle duality and if the particle nature only is considered then the geometrical optics results. In the next two sections the theoretical PSFs assuming geometrical optics and diffraction-limited optics are obtained for comparison with experimental results.

3.3.2 Geometrical Optics Approach

Pentland [3] showed that for the simple optical system shown in Figure 3.3 and assuming geometrical optics the PSF is a pillbox with a blur circle radius given by

$$r = \frac{v_0 D - F(v_0 + D)}{f D} \tag{3.8}$$

where v_0 is the distance between the lens and the CCD, D is the depth of the object (denoted u in the figure), F is the focal length of the lens and f is the f-number, which is defined as $f = \frac{2F}{d}$. The distance u_0 is the distance at which an object would appear in focus on the image plane.

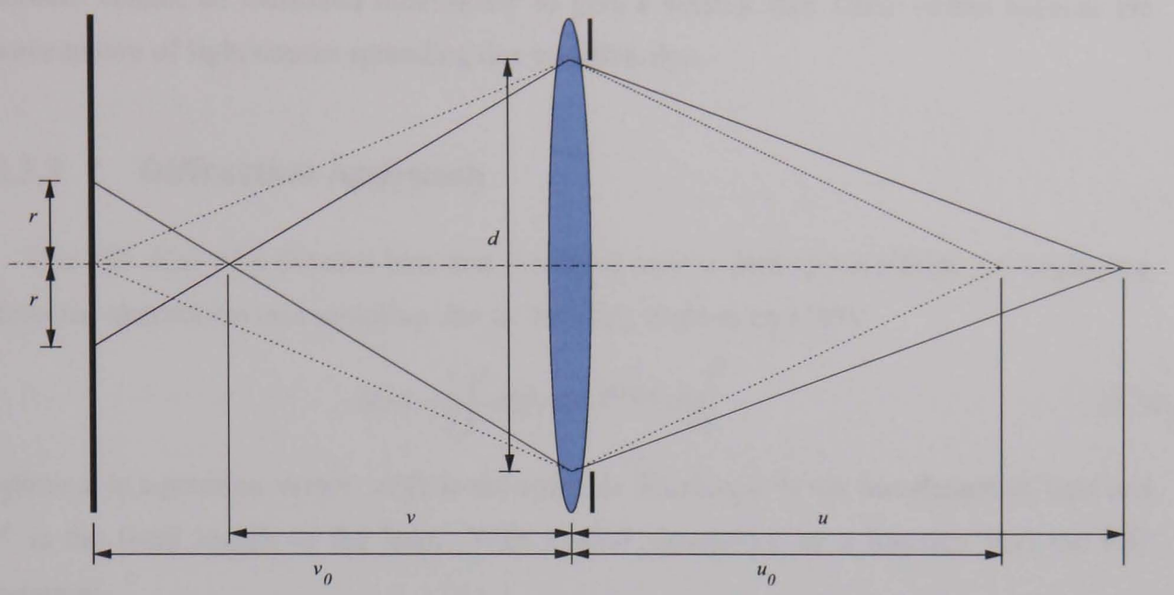


Figure 3.3: A simple model of the optical system with the image plane on the left-hand side

Figure 3.4 shows a plot of the blur circle radius where $F = 16\,\text{mm}$ and $v_0 = 16.57\,\text{mm}$ for three different aperture sizes. Under geometrical optics the PSF is the same shape as the aperture and for a circular aperture the PSF is a pillbox (or cylindrical) function. Geometrical optics neglect the wave-nature of electromagnetic radiation and thus the results are independent of the wavelength.

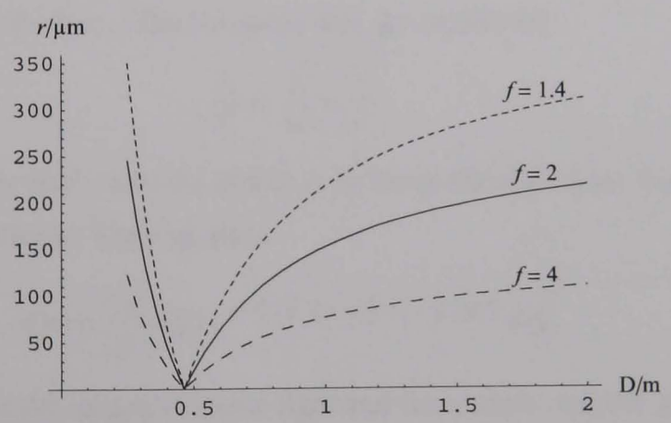


Figure 3.4: Blur circle radius r as a function of depth D for three f-numbers

Note from Figure 3.4 that for a given blur circle radius r and aperture f there are two corresponding depths and the ambiguity can be resolved by setting the object to be either in front or behind the point of focus.

The maximum blur circle radius assuming geometrical optics is found from

$$\lim_{D\to\infty} \frac{v_0 D - F(v_0 + D)}{f D} = \frac{v_0 - F}{f} \tag{3.9}$$

and clearly for a given focus position v_0 and fixed focal length F as the f-number is increased (i.e. smaller aperture) the maximum blur circle radius decreases. In reality the f-

number cannot be increased indefinitely to give a smaller blur circle radius because the wave nature of light causes spreading due to diffraction.

3.3.3 Diffraction Approach

The PSF $h(\boldsymbol{x})$ of a focused lens that is subject only to diffraction effects, i.e. neglecting defocus, aberrations and sampling due to the FPA, is given by [123]

$$h(\boldsymbol{x}) = \left| \int A(\boldsymbol{\xi})\, e^{-j\frac{2\pi}{\lambda F}\boldsymbol{\xi}\cdot\boldsymbol{x}}\, d\boldsymbol{\xi} \right|^2 \tag{3.10}$$

where $\boldsymbol{x}$ is a position vector, $A(\boldsymbol{\xi})$ is the aperture function, λ is the wavelength of light and F is the focal length of the lens. With optical aberrations as a function $\theta(\boldsymbol{x})$ the PSF becomes

$$h(\boldsymbol{x}) = \left| \int A(\boldsymbol{\xi})\, e^{j\,\theta(\boldsymbol{\xi})}\, e^{-j\frac{2\pi}{\lambda F}\boldsymbol{\xi}\cdot\boldsymbol{x}}\, d\boldsymbol{\xi} \right|^2 . \tag{3.11}$$

Out-of-focus blurring can be modelled as a quadratic aberration of the form [123]

$$\theta(\boldsymbol{x}) = \frac{\pi}{\lambda}\left(\frac{1}{u} + \frac{1}{v} - \frac{1}{F}\right)|\boldsymbol{x}|^2 \tag{3.12}$$

where u is the distance between the object and the lens, and v is distance between the screen (or FPA) and the lens. The Gaussian lens law states that

$$\frac{1}{F} = \frac{1}{u} + \frac{1}{v} \tag{3.13}$$

and so it can be seen that when the object is in focus the aberration $\theta(\boldsymbol{x})$ reduces to zero. Substituting in the defocus blurring gives

$$h(\boldsymbol{x}) = \left| \int A(\boldsymbol{\xi})\, e^{-j\frac{\pi}{\lambda}\left(\frac{2}{F}\boldsymbol{\xi}\cdot\boldsymbol{x} - \left(\frac{1}{u}+\frac{1}{v}-\frac{1}{F}\right)|\boldsymbol{\xi}|^2\right)}\, d\boldsymbol{\xi} \right|^2 \tag{3.14}$$

Equation (3.14) is for monochromatic light and for a more realistic analysis for DFD it is assumed that the PSF is due to polychromatic light that is white, i.e. it is of constant intensity, between the wavelengths of λ_1 and λ_2. The PSF is then given by

$$h(\boldsymbol{x}) = \int_{\lambda_1}^{\lambda_2} \left| \int A(\boldsymbol{\xi})\, e^{-j\frac{\pi}{\lambda}\left(\frac{2}{F}\boldsymbol{\xi}\cdot\boldsymbol{x} - \left(\frac{1}{u}+\frac{1}{v}-\frac{1}{F}\right)|\boldsymbol{\xi}|^2\right)}\, d\boldsymbol{\xi} \right|^2 d\lambda \tag{3.15}$$

and if the PSF is assumed to be a 1D function then the vector $\boldsymbol{x}$ becomes the scalar x and the vector $\boldsymbol{\xi}$ becomes ξ and so

$$h(x) = \int_{\lambda_1}^{\lambda_2} \left| \int A(\xi)\, e^{-j\frac{\pi}{\lambda}\left(\frac{2}{F}\xi x - \left(\frac{1}{u}+\frac{1}{v}-\frac{1}{F}\right)|\xi|^2\right)}\, d\xi \right|^2 d\lambda . \tag{3.16}$$

If the aperture function $A(\xi)$ is assumed to be a circle then in 1D it forms a slit with

$$A(\xi) = \begin{cases} 1 & \xi \le r \\ 0 & \xi > r \end{cases} \tag{3.17}$$

and so the PSF becomes

$$h(x) = \int_{\lambda_1}^{\lambda_2} \left| \int_{-r}^{r} e^{-j\frac{\pi}{\lambda}\left(\frac{2}{F}\xi x - \left(\frac{1}{u}+\frac{1}{v}-\frac{1}{F}\right)|\xi|^2\right)} d\xi \right|^2 d\lambda. \tag{3.18}$$

Figure 3.5 shows the PSF expected for a focused scene where only diffraction is present. The ideal monochromatic light is taken as 700 nm, corresponding to red light and the polychromatic light is taken as an ideal white light source with equal intensity components in the range 400 to 700 nm. Note the similarity between the PSF due to diffraction effects and the Gaussian function.

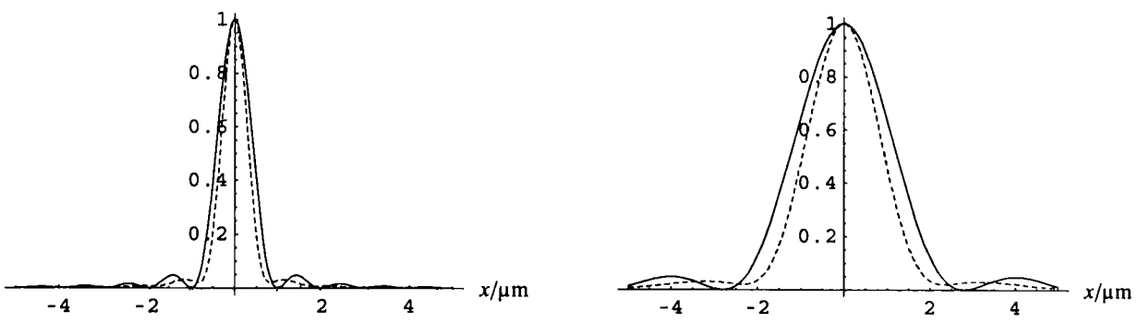


Figure 3.5: PSFs for focused monochromatic (solid) and polychromatic (dashed) light for f-numbers of 1.4 (left) and 4 (right)

Figures 3.6 and 3.7 shows the theoretical PSF for a defocused 16mm lens where the camera is focused at 0.464m and the point source is at 0.8m and 0.6m. The effect of the polychromatic light is to smooth out the PSF and make it look more like a pillbox function. Note that the ringing is not caused by noise in the processing, but instead the ripple effect due to diffraction of light, where its wave nature has been taken into account.

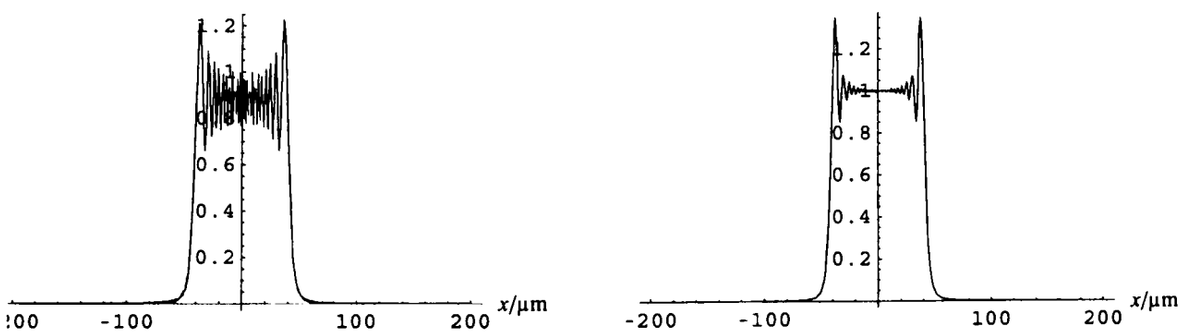


Figure 3.6: PSFs for defocused monochromatic (left) and polychromatic (right) light for a defocused system with a depth of 0.6m

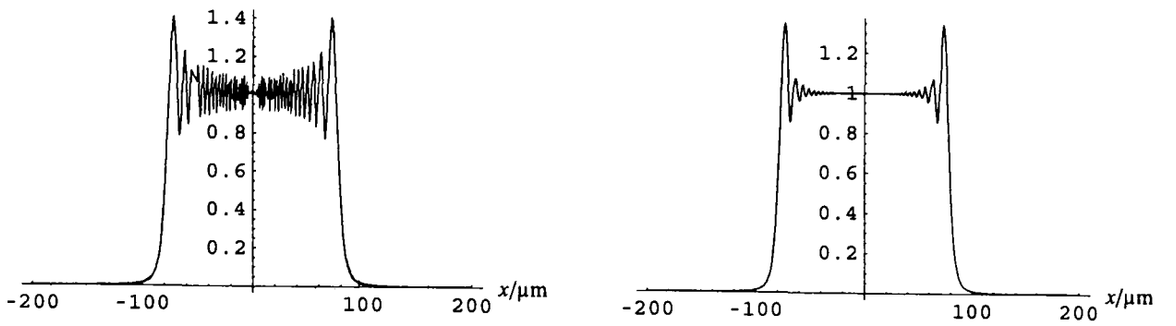


Figure 3.7: PSFs for defocused monochromatic (left) and polychromatic (right) light for a defocused system with a depth of 0.8m

A Gaussian was fitted to the PSFs and the standard deviation was plotted as a function of distance in Figure 3.8. Note that a consequence of the diffraction is that near the focus position the wider aperture (f/1.4) has a narrower PSF and thus a smaller standard deviation as shown in the right hand plot of Figure 3.8.

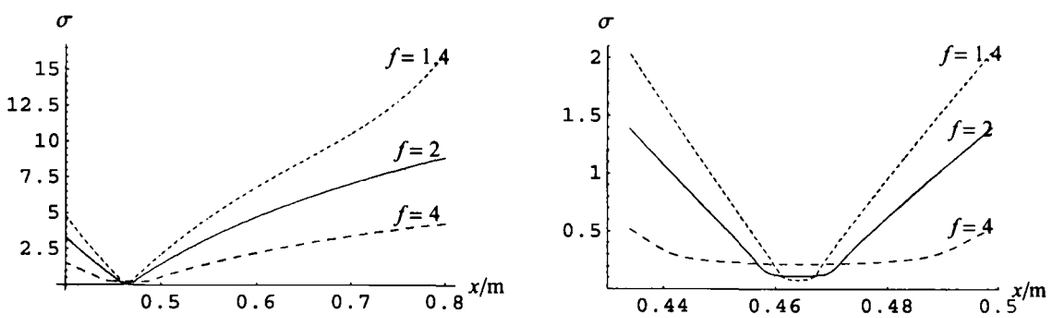


Figure 3.8: (Left) Standard deviation σ of a fitted Gaussian as a function of depth; (Right) Zoomed in version around the focus position at 0.464m

3.3.4 Conclusion

This section has examined theoretical PSFs assuming geometrical and diffraction-limited optics. Under geometrical optics the PSF due to defocus blurring takes the shape of the aperture, which is a pillbox for a circular aperture. When the lens is focused the PSF becomes a delta function, having infinite intensity and infinitesimal spatial extent. Diffraction optics takes into account the wave-nature of light and the shape changes with increasing defocus from a function that approximates a Gaussian to one that resembles a pillbox.

3.4 Theoretical Edge Spread Functions

3.4.1 Introduction

The Point Spread Function can be slowly built up from the PRF or instead using any of the average PSF producing techniques described in Section 3.2.2. In this research the knife-edge technique was employed and this section firstly considers an improvement to Staunton's [57] algorithm that incorporates the effect of non-uniform illumination of the lightbox. Without noise the ESF could be differentiated to yield the PSF, but differentiating a noisy function amplifies the noise. Models of the ESF were developed assuming particular shapes of the PSF and a regularised numerical differentiation process was proposed.

3.4.2 Non-Uniform Illumination Considerations

An ideal brightness step would consist of two regions of different brightnesses separated by an abrupt transition and within each region the brightness would be constant, as shown by the dashed line in Figure 3.9. Experimentally a light box can be employed with a knife-edge to approximate a step edge, however it is not necessarily the case that the brightnesses of the regions are uniform. If this non-uniformity is not taken into account the resultant PSFs will be erroneous and so its effect must be eliminated as much as possible for accurate measurements. Instead of assuming a constant brightness, the model was changed to consist of the abrupt transition as before, but each region can have a linear change in intensity as a function of spatial position. As the bulbs are near the edge of the lightbox the intensity could drop towards the centre, hence the positive gradient of the upper region. The intensity of the lower region is due to the ambient light reflecting off the darker area of the lightbox, which will be dimmer than that due to the bulbs.

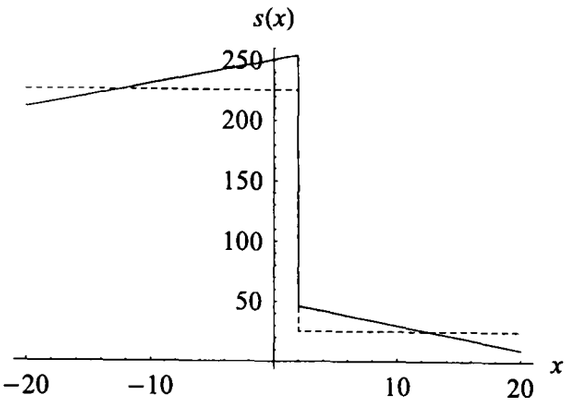


Figure 3.9: A model of the ideal step without (dashed line) and with (solid line) non-uniform illumination

An ideal step $s(x)$ incorporating the non-uniform illumination improvement with an abrupt transition at $x = x_0$ is given by

$$s(x) = (m_1 x + c_1)\, u(x + x_0) + (m_2 x + c_2)\, u(x - x_0) \tag{3.19}$$

where $u(x)$ is a unit step function given by

$$u(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \tag{3.20}$$

and c_1 and c_2 are the brightnesses of the upper and lower intensity regions and m_1 and m_2 are the gradients of the brightnesses. In the original case assuming uniform illumination $m_1 = m_2 = 0$.

If there was no noise, sampling, diffraction or defocus effects then the camera would return a profile like that in Figure 3.9 with the solid line, however, those effects are present in a real system. The deviation of the ESF from the ideal model shown can be used to determine the PSF of the complete camera system. The next sections consider different models of the PSF and their subsequent ESFs.

3.4.3 Pillbox PSF

Under geometrical optics assumptions the PSF due to defocus is a pillbox, which is given by

$$h_p(x) = \frac{1}{2\sigma}[u(x + \sigma) - u(x - \sigma)] \tag{3.21}$$

where σ is the radius of the pillbox, and hence the blur circle. An example of the pillbox is shown in Figure 3.10 for $\sigma = 5$pixels.

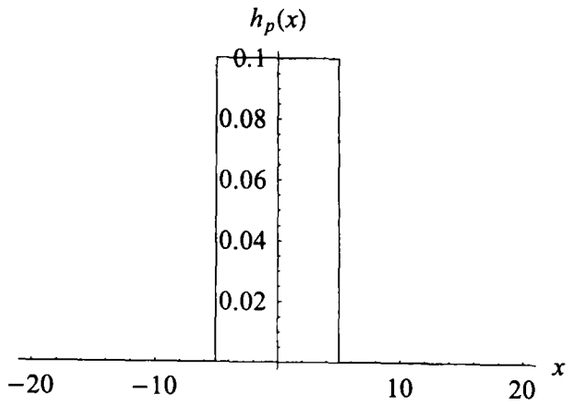


Figure 3.10: The pillbox PSF with a radius $\sigma = 5$

The ESF assuming a pillbox PSF and an ideal step incorporating non-uniform illumination is derived in Appendix A and the result is

$$f_p(x) = \begin{cases} m_1 x + c_1 & x - x_0 < -\sigma \\ \frac{1}{4\sigma}[-(2c_1 + m_1(x + x_0 - \sigma))(x - x_0 - \sigma) + & -\sigma \le x - x_0 \le \sigma \\ \quad (x - x_0 + \sigma)(2c_2 + m_2(x + x_0 + \sigma))] & \\ m_2 x + c_2 & \sigma < x - x_0 \end{cases} \tag{3.22}$$

where x_0 is the location of the abrupt transition and m_1, m_2, c_1 and c_2 are the parameters of the linear segments. An example of the ESF for a PSF with a blur circle radius $\sigma = 5$ is shown in Figure 3.11 where the original step is shown with a dotted line for comparison.

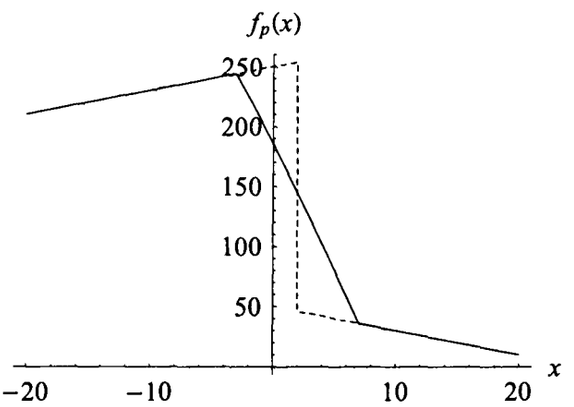


Figure 3.11: ESF with a pillbox PSF where $\sigma = 5$ (solid line) and the ideal step edge (dashed line)

Note that there are two very sharp transitions in the ESF. A pillbox PSF would result if the lens passed every spatial frequency, however, due to diffraction effects it is known that this is not possible. Although the pillbox PSF does not appear to be physically realisable it has been kept for comparison purposes.

3.4.4 ESF Modelled as a Sum of Fermi-Dirac Functions

The Fermi-Dirac distribution is important in quantum physics for giving the probability that an electron occupies a particular energy state E and it is given by

$$P_{\mathrm{FD}}(E) = \frac{1}{1 + \exp\left\{\frac{E - E_F}{kT}\right\}} \tag{3.23}$$

where E_F is the Fermi-level, k is Boltzmann's constant and T is the temperature [124]. The shape of the PDF resembles that of a defocused step edge, but in order to allow a better fit, Tzannes and Mooney [56] fitted a sum of three Fermi-Dirac functions to the ESF. At the Fermi-level $E = E_F$ the probability $P_{\mathrm{FD}}(E) = \frac{1}{2}$ and thus E_F is the centre point of the ESF. The sum of N Fermi-Dirac functions for modelling the ESF can be in a general form as

$$f(x) = \sum_{i=1}^{N} \left(\frac{a_i}{b_i + \exp\left\{\frac{x - c_i}{d_i}\right\}} + e_i \right) \tag{3.24}$$

where the constants a_i have been added to ensure the intensity can exceed unity and the e_i terms account for the non-zero brightness of the lowest level. An example of the ESF produced assuming a Fermi-Dirac function is shown in Figure 3.12.

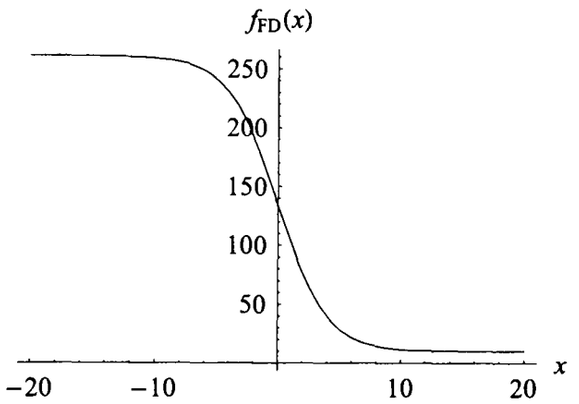


Figure 3.12: ESF with a Fermi-Dirac PSF

In order to recover the PSF the ESF must be differentiated, which is given by

$$h_{\mathrm{FD}}(x) = -\frac{\partial f(x)}{\partial x} = -\sum_{i=1}^{N} \left(\frac{a_i \exp\left\{\frac{x - c_i}{d_i}\right\}}{d_i \left(b_i + \exp\left\{\frac{x - c_i}{d_i}\right\}\right)^2} \right) \tag{3.25}$$

and an example PSF is shown in Figure 3.13.

The main problem with the Fermi-Dirac function is that it does not take into account the non-uniform illumination in a way that allows the step and the PSF to be separated. To achieve this a new mathematical formulation would be required, but it would no longer be simply a sum of Fermi-Dirac functions as used by Tzannes and Mooney [56], which has been employed for comparison purposes with earlier work.

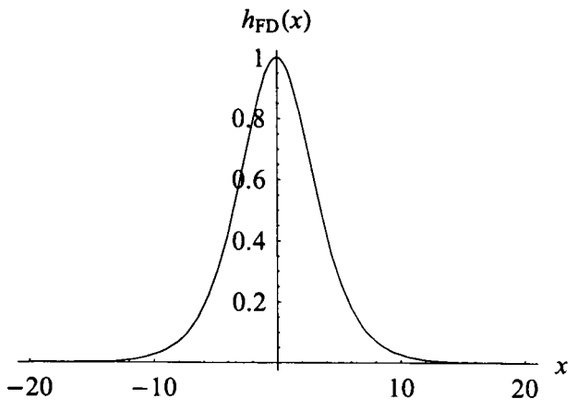


Figure 3.13: Fermi-Dirac PSF

3.4.5 Gaussian PSF

The Gaussian PSF is the most frequently assumed model found in literature on depth-from-defocus and this is partly due to its simplicity. A one-dimensional Gaussian with a standard deviation σ and centred at $x = x_0$ is given by

$$h_g(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left\{-\frac{1}{2}\,\frac{(x-x_0)^2}{\sigma^2}\right\} \tag{3.26}$$

and an example of the Gaussian is shown in Figure 3.14.

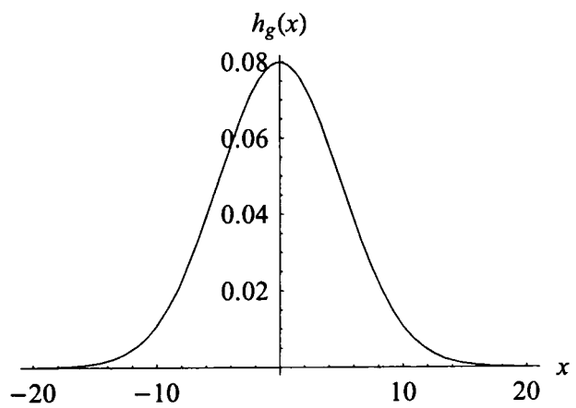


Figure 3.14: Gaussian PSF with $\sigma = 5$ and $x_0 = 0$

As shown in Appendix A, the ESF assuming a Gaussian PSF and a step edge with non-uniform illumination is given by

$$f_g(x) = \frac{1}{2}\left[-(m_1 + m_2)\,\sigma\sqrt{\frac{2}{\pi}}\; e^{-\frac{1}{2}\frac{(x-x_0)^2}{\sigma^2}} + \right.$$
$$\left.(m_1 x + c_1)\left(1 - \operatorname{erf}\left(\frac{x-x_0}{\sigma\sqrt{2}}\right)\right) + (m_2 x + c_2)\left(1 + \operatorname{erf}\left(\frac{x-x_0}{\sigma\sqrt{2}}\right)\right)\right] \tag{3.27}$$

where $\operatorname{erf}(\cdot)$ is the error function, defined as

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt. \tag{3.28}$$

If the ideal step with non-uniform illumination as shown in Figure 3.9 is defocused with the Gaussian shown in Figure 3.14 then the ESF is as shown in Figure 3.15.

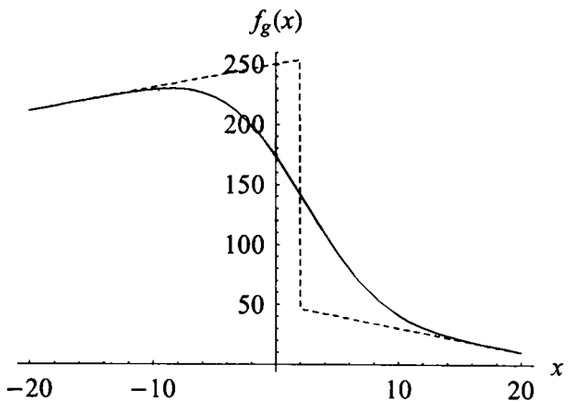


Figure 3.15: ESF when the PSF is a Gaussian with $\sigma = 5$ (solid line) and the ideal step edge (dashed line)

3.4.6 Generalised Gaussian PSF

The Generalised Gaussian function is a novel model being proposed for the PSF of a defocused lens. Along with the mean $\hat{x}$ and the standard deviation σ, the power p of the function is required. The function can take the form of a Gaussian when the power $p = 2$ and a pillbox when $p = \infty$, and thus encompasses both of the frequently used models of defocus. The Generalised Gaussian is given by

$$h_G(x) = \frac{p^{1-\frac{1}{p}}}{2\sigma\,\Gamma\left(\frac{1}{p}\right)} \exp\left\{-\frac{1}{p}\,\frac{|x-\hat{x}|^p}{\sigma^p}\right\} \tag{3.29}$$

where $\Gamma(\cdot)$ is the Gamma function and $|\cdot|$ represents the modulus. The term before the exponential ensures the function has unit area. Two Generalised Gaussian functions are presented in Figure 3.16 for ($p = 1$, $\sigma = 5$) and ($p = 4$, $\sigma = 5$).

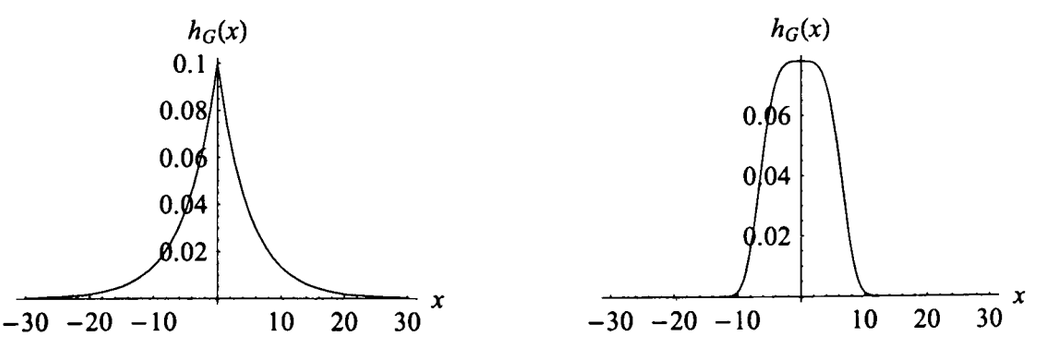


Figure 3.16: Generalised Gaussian PSFs where (left) ($p = 1$, $\sigma = 5$) and (right) ($p = 4$, $\sigma = 5$)

The ESF assuming a step edge with non-uniform illumination and a Generalised Gaussian PSF is given by the convolution of the former function with the latter. A closed form, algebraic solution could not be obtained with Mathematica or Maple packages and so the convolution integral must be evaluated numerically. The ESF is given by

$$f_G(x) = \frac{p^{1-\frac{1}{p}}}{2\sigma\Gamma\left(\frac{1}{p}\right)} \int_{x-x_0}^{\infty} \exp\left\{-\frac{1}{p}\frac{|\xi|^p}{\sigma^p}\right\}[m_1(x-\xi)+c_1]\,d\xi + \frac{p^{1-\frac{1}{p}}}{2\sigma\Gamma\left(\frac{1}{p}\right)} \int_{-\infty}^{x-x_0} \exp\left\{-\frac{1}{p}\frac{|\xi|^p}{\sigma^p}\right\}[m_2(x-\xi)+c_2]\,d\xi \tag{3.30}$$

as shown in Appendix A. Using the PSFs shown in Figure 3.16 and the ideal step with non-uniform illumination the resulting ESFs are shown in Figure 3.17.

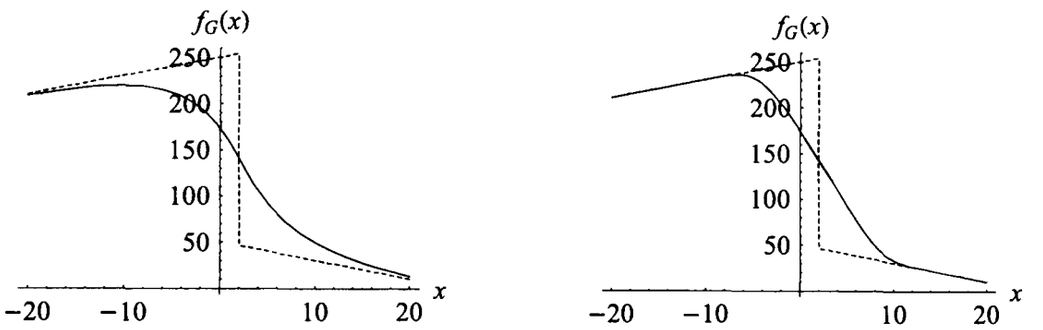


Figure 3.17: The ideal steps (dashed lines) and the ESFs (solid lines) assuming Generalised Gaussian PSFs with (left) $p = 1$ and $\sigma = 5$; (right) $p = 4$ and $\sigma = 5$

Note that the higher the power of the Generalised Gaussian the sharper the transition points until in the limit the PSF is a pillbox and then an ESF like that shown in Figure 3.11 is produced.

3.4.7 Regularised Numerical Differentiation

In order to recover the PSF from the super-resolution Edge Spread Function (ESF) the response must be differentiated and as the data is discrete finite-difference approximations must be employed. Consider the problem of finding the derivative of a function $f(x)$ where x is a discrete variable taking integer values. A simple approximation to the derivative is given by the *forward difference formula*

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} \tag{3.31}$$

where the spacing between the samples $x_{i+1} - x_i$ is sufficiently small and the function $f(x)$ is smooth. If the same data has been corrupted by additive white Gaussian noise (AWGN) so that each observation is given by $g(x_i) = f(x_i) + \varepsilon_i$ then the derivative of the observed data $g(x)$ is now given by

$$g'(x_i) = \frac{g(x_{i+1}) - g(x_i)}{x_{i+1} - x_i} = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} + \frac{\varepsilon(x_{i+1}) - \varepsilon(x_i)}{x_{i+1} - x_i}. \tag{3.32}$$

It is assumed that the underlying function $f(x)$ is smooth thus ensuring that the gradient of the noise is significant due to its lack of correlation.

The five-point numerical differentiation formula is given by [125]

$$f'(x_i) = \frac{f(x_{i-2}) - 8f(x_{i-1}) + 8f(x_{i+1}) - f(x_{i+2})}{12} \tag{3.33}$$

and although it only uses four points, it is derived from the Lagrange polynomials for five points. It is more accurate than the two-point formula and helps to reduce the noise more, however, experimentally the results were very poor, as shown in Section 4.5.4.

Chartrand considered the problem of finding the derivative of a function when the underlying function is noisy and has a discontinuity in the derivative. The solution proposed uses total-variation regularisation where the derivative of a function $f(x)$ defined on the closed interval $[0, L]$ is the minimiser of the function [126]

$$F(u) = \alpha \int_0^L |u'(x)|\,dx + \frac{1}{2}\int_0^L \left|\left(\int_0^x u(y)\,dy\right) - f(x)\right|^2 dx \tag{3.34}$$

where $u(x)$ is the first derivative of the function $f(x)$ and α is a regularisation term that weights the penalty term

$$\int_0^L |u'(x)|\,dx \tag{3.35}$$

against the data fidelity term

$$\frac{1}{2}\int_0^L \left|\left(\int_0^x u(y)\,dy\right) - f(x)\right|^2 dx. \tag{3.36}$$

The total variation suppresses the noise without removing discontinuities in the derivative [126]. The appeal of this approach is that a pillbox PSF has a two finite discontinuities and this method ensures that they can be recovered and additionally noise suppression is achievable. Gradient descent could be used to find the optimum u but convergence is slow, thus Chartrand used lagged-diffusivity [126] and implemented the algorithm in MATLAB. The main problem is that the choice of the regularisation parameter α affects the derivative produced.

3.4.8 Conclusion

In this section the step in intensity incorporating non-uniform illumination has been proposed as a better model of the ideal knife-edge. Further, different ESF models have been proposed with their associated PSFs. In particular the Generalised Gaussian, a novel model for the PSF of a defocused lens system, was proposed. The regularised numerical differentiation eliminates the requirement of a model of the PSF, however, a regularisation term α must be chosen, which affects the subsequent shape of the PSF.

3.5 Conclusion

In this chapter a variety of PSF and OTF measurement techniques have been presented ranging from PRF methods that use a sub-micron precision source to sinusoidal targets and knife-edge methods. The theoretical PSFs assuming geometrical and diffraction-limited optics have been examined and in particular the PSF appears to change from an approximately Gaussian shape to that of a pillbox with increasing defocus.

One of the PSF measurement methods presented employs a lightbox and a knife-edge from which the ESF can be measured. The effect of non-uniform illumination has been discussed as an important improvement and ESFs assuming pillbox, Gaussian and Generalised Gaussian PSF models have been developed, the latter being a novel solution. The regularised numerical differentiation has been suggested as another technique, but the problem then becomes finding the required regularisation parameter.

In the next chapter the results of performing experiments on a real camera system are presented and the techniques discussed in this chapter are applied.

Chapter 4

The Results from the Measurement of the Point Spread Function of a Defocused Imaging System

4.1 Introduction

The literature review of PSF and MTF measurement techniques illustrated that there are many different ways of characterising an optical system. The most accurate way of determining a PSF is undoubtedly using a point source that can be moved at the sub-micron level to build up a PRF, but it would be an extremely time-consuming method. For DFD work it is important that an average PSF is produced for polychromatic light, which rules out the laser-based techniques, such as interference gratings. The knife-edge technique developed by Reichenbach *et al.* [55] and improved by Tzannes and Mooney [56] and Staunton [57] showed good results and was the basis for the experimental work.

The PSF is a function of the camera parameters and the depth of the object and it is important to measure the PSF for different camera settings and depths. Either the lightbox or the camera could be moved and a computer-controlled x-stage was built to move the camera in the required small increments.

Uniform lightbox illumination is required for existing techniques and the increased spatial extent of the PSF due to defocusing caused experimental difficulties as the assumption did not hold. To solve the problem, the ESF fitting algorithm was improved to incorporate non-uniform illumination, the parameters of which were found automatically.

A Gaussian has long been used as a model of the PSF of a lens system and the Generalised Gaussian function is proposed as a better model as it can encompass both the Gaussian and pillbox shapes and mixtures of the two, with a cost of increased complexity. The super-resolution ESF must be differentiated to recover the PSF and this chapter shows the

application of a regularised numerical differentiation method developed by Chartrand [126].

The linearity and noise experiments are presented in Sections 4.2 and 4.3. The hardware built to move the camera in small increments is described in Section 4.4 followed by the PSF recovery algorithm that processes the images produced by the camera in Section 4.5. One-dimensional PSF results are presented in Section 4.5.4 to illustrate the advantages and disadvantages of the different PSF models. Results for a 16mm video and a 24mm Sigma photographic lens are presented in Sections 4.6 and 4.7 respectively and in particular, 2D PSFs for the Sigma lens are given in Section 4.7.5. Finally, the findings are summarised in Section 4.8.

4.2 Linearity Experiments

4.2.1 Introduction

It is important in DFD and PSF measurement work that the camera produces a linear response to light intensity and Section 4.2.2 outlines some methods for measuring the linearity. A circuit was devised to measure the intensity of an LED using a photodiode and the response of the camera measured, as shown in Section 4.2.3. The results in Section 4.2.4 show the output of the camera as the brightness of the LED is changed in small steps.

4.2.2 Methods of Measurement

A common assumption in DFD and PSF measurement algorithms is that the camera is a linear system with the property that

$$T(c_1 I_1 + c_2 I_2) = c_1 T(I_1) + c_2 T(I_2) \tag{4.1}$$

where $T(\cdot)$ is the transfer function, c_1 and c_2 are constants and I_1 and I_2 are overlapping image regions. It was important to test this linearity assumption and if it fails to produce a look-up table to compensate.

The operation of a CCD was discussed in Section 3.2 and in particular note how a single photon produces a single photoelectron, thus suggesting linearity. However, the non-linearity in the charge accumulation occurs as the potential well develops a negative charge, which repels further electrons. The charge can leak into adjacent pixels leading to a process called *blooming* or *bleeding*. Further, the output amplifier and ADC cannot

accurately count the charge above the *saturation level*, leading to another cause of non-linearity. The non-linear properties of CRTs means that gamma correction [127] is often applied in cameras using dedicated hardware and this needs to be turned off.

There are various methods for measuring the linearity of the camera including:

- A pulsed LED can be used where the mark-to-space ratio can be changed, but the main problem is that the camera must be synchronised with the LED [24]
- The integration time of the camera can be varied. The number of electrons in a photo-site is given by [111]

$$I = T \int_{\lambda}\int_{y}\int_{x} B(x, y, \lambda)\, S_r(x, y)\, q(\lambda)\, dx\, dy\, d\lambda \tag{4.2}$$

  where T is the integration time (seconds), $B(x, y, \lambda)$ is the incident spectral irradiance (W m^{-2}) at position (x, y), $q(\lambda)$ is an efficiency term (electrons J^{-1}) as a function of wavelength and $S_r(x, y)$ is the spatial response of the photosite. Thus it can be seen that the number of electrons is directly proportional to the integration time.
- Neutral density filters or liquid absorption standards can be employed [24] or two polarising filters could be set up where the relative angle is used to vary the intensity.
- The current through the LED can be changed to alter its brightness, however, the response is not linear with applied current. By measuring the brightness of the LED and cross-referencing it with the output of the camera the linearity can be measured.

It was decided that the final proposal would be employed because this solution removes the problem with the pulsed LED approach concerning the synchronisation and it does not require expensive neutral density filters. The linearity of the integration time setting on the camera could not be assumed and thus it seemed that more precise experiments could be performed by changing the LED's brightness.

4.2.3 The Linearity Measurement Circuit Devised

It was decided that three LEDs (red, green and blue to span the spectral range) would be used in turn and the current through the LED was varied to give the required brightness. A transimpedance amplifier converted the current through the photodiode to a voltage and then a low-pass filter stage was used to reduce the shot and thermal noise. A photodiode was employed because it is known that a photodiode's current is linearly related to light intensity and thus the output voltage was linearly related to the light intensity of the LED.

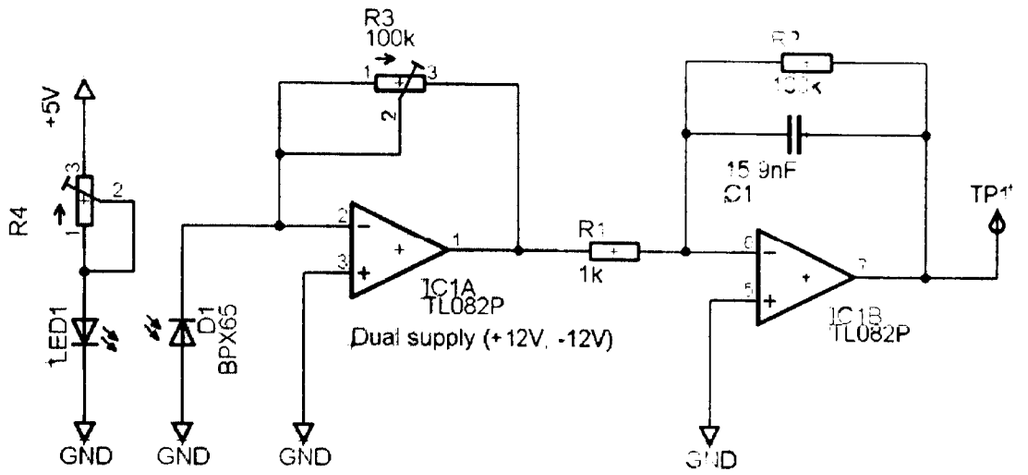


Figure 4.1: The emitter and detector circuit devised for the linearity measurement

4.2.4 Results

The Basler A631fc colour camera was focused on the LED and ten images were taken per voltage reading, so that an average could be taken to reduce noise. A MATLAB script read in the images and the maximum pixel intensity in the required colour plane was plotted against voltage. The red, green and blue LEDs had peak wavelengths of 700nm, 565nm and 488nm respectively. An infra-red (IR) LED with a peak wavelength of 1000nm was imaged, but the camera could not detect the light until too much current was applied and diode combustion was viewed; thus indicating that the IR cut-off filter that only transmits light in the range 400-720nm [26] was working effectively.

Figures 4.2 to 4.4 show the results for the red, green and blue colour planes respectively. The camera was saturated when the intensity reached the level of 255 as it has an 8-bit analogue-to-digital converter (ADC). The minimum pixel intensity does not go to zero due to the noise processes discussed in the next section. Note that the change of the x-axis scales is due to differences in the LEDs and the quantum efficiency of the photodiode as a function of wavelength.

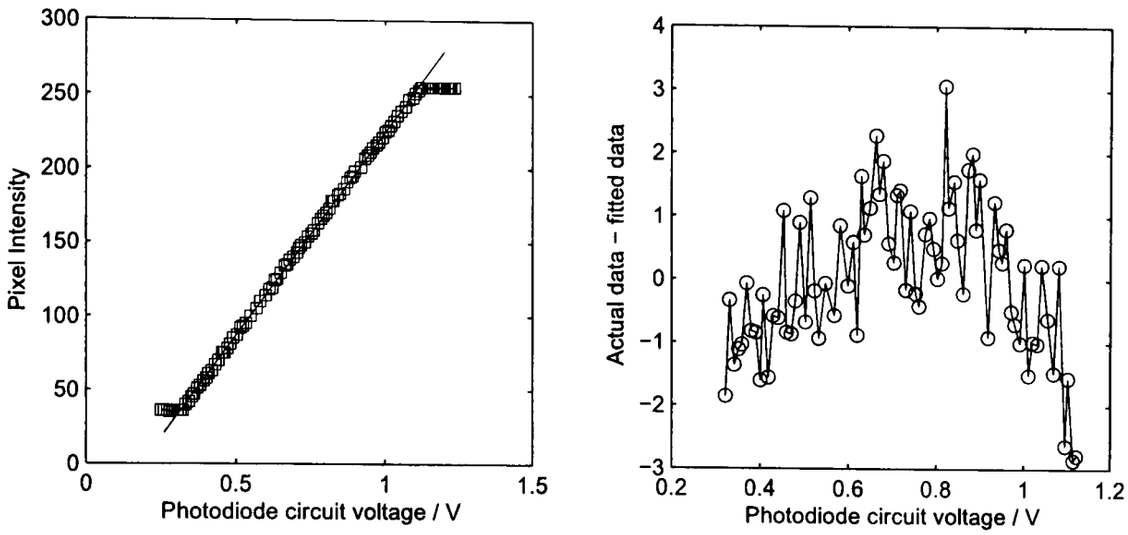


Figure 4.2: Red LED linearity experiment (r = 0.9997 and MSE = 1.4129)

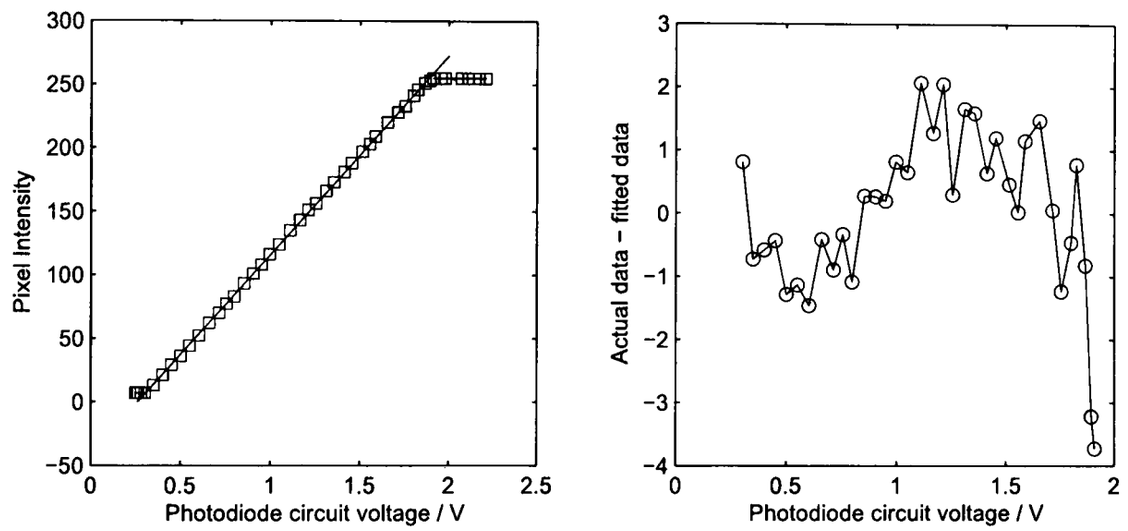


Figure 4.3: Green LED linearity experiment (r = 0.9997 and MSE = 1.6757)

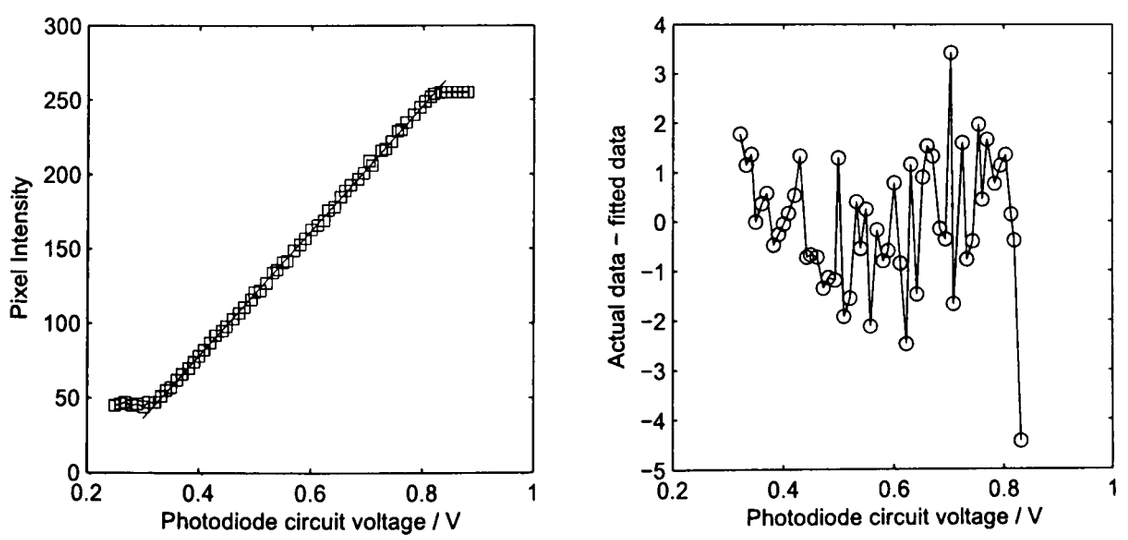


Figure 4.4: Blue LED linearity experiment (r = 0.9996 and MSE = 1.7930)

The left-hand image for a given colour plane shows the pixel intensity as a function of the photodiode circuit voltage, which is proportional to brightness of the LED. The right-hand image shows the residuals, defined as the difference between the actual data and the fitted line, so that the structure can be ascertained. The correlation coefficient r gives a measure of the fit of the experimental data to a straight line and it is given by [128]

$$r = \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{Var}(X)}\,\sqrt{\mathrm{Var}(Y)}} \tag{4.3}$$

where it is always the case that $-1 \le r \le 1$. The covariance of X and Y is denoted $\mathrm{Cov}(X, Y)$ and is given by

$$\mathrm{Cov}(X, Y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})\,(y_i - \overline{y}) \tag{4.4}$$

where variables X and Y have N elements, the i^{th} elements are denoted x_i and y_i respectively and $\overline{x}$ and $\overline{y}$ are the mean values of X and Y given by

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{4.5}$$

and

$$\overline{y} = \frac{1}{N}\sum_{i=1}^{N} y_i. \tag{4.6}$$

The variance $\mathrm{Var}(X)$ is given by

$$\mathrm{Var}(X) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \overline{x})^2. \tag{4.7}$$

and similarly for $\mathrm{Var}(Y)$. For a perfect fit the correlation coefficient $r = 1$ and the values for each of the colour planes in Figures 4.2 to 4.4 are very close to a perfect fit. Only the data in the linear region was used to fit the line and calculate the correlation coefficient. The results clearly show that the camera has an output that is a linear function of brightness except near the saturation region close to intensities of 255 and the lowest brightness is not zero due to noise offsets, as discussed in Section 4.3. Thus, the experiments confirmed that gamma correction was not applied by the camera.

4.2.5 Conclusion

The circuit designed produces a voltage that is linearly related to the intensity of an LED under test from which the linearity of the camera can be measured. The results presented above for each of the colour planes of the Basler A631fc colour camera show that the linearity of the camera was very good in the working range with a correlation coefficient that is essentially unity.

4.3 Noise Experiments

4.3.1 Introduction

Noise is an inescapable property of an imaging system and the different processes were discussed in Section 1.3.4. In the next section the bias and dark frame measurements are presented for each of the colour planes.

4.3.2 Bias and Dark Frame Measurements

The dark noise can be ascertained from a *dark frame*, which is an image with the lens cap on essentially and taken using the maximum shutter time, which was 8.19 ms for the Basler A631fc colour camera employed. One thousand images were taken with the camera and the mean and variance of each pixel was calculated.

The readout noise can be measured from a bias frame, which should be taken with a zero duration exposure time, but that is generally not possible unless access to the circuitry can be obtained. In order to approximate this setting the shortest exposure of 20 μs was used. The mean intensities of the two measurements for each colour plane are presented in Table 4.1.

Table 4.1. Mean pixel intensity of the colour planes for two noise tests

Colour Plane	Bias frame	Dark frame
Red	31.68	31.70
Green	15.93	15.94
Blue	31.86	31.86

4.3.3 Analysis of the Measurements

Considering there is a factor of 4095 between the exposure time for the bias and dark frames the results are very similar, suggesting that the integration time has very little effect on the noise level. When the CCD is not exposed to any light the only electrons in the photosite are thermally generated, i.e. not by incident photons creating electron-hole pairs. As the integration time increases the number of thermally generated electrons also increases and if the thermal noise dominated then a longer shutter time would produce a higher mean noise level. Thus, the main contribution to the noise was due to the read-out electronics.

4.3.4 Offset Subtraction

The ADC and VGC frequently employ an offset brightness for electronic design reasons and it is very important that this offset is subtracted. Assume that the linear region of the response is used and the radiance of a point, denoted x, is related to the quantised brightness level of the camera y by

$$y = m\,x + c \tag{4.8}$$

where m is the gain and c is the offset, which cannot be assumed to be zero. Now consider the ratio of the brightness of two colour planes y_1 and y_2 given by

$$\frac{y_1}{y_2} = \frac{m_1\,x + c_1}{m_2\,x + c_2}. \tag{4.9}$$

The ratio changes with the actual brightness x, which is an unwanted effect. If the offset of each colour plane is subtracted then the ratio becomes

$$\frac{y_1 - c_1}{y_2 - c_2} = \frac{m_1\,x}{m_2\,x} = \frac{m_1}{m_2} \tag{4.10}$$

and thus ensuring that the ratio remains constant.

4.4 The Automation Hardware

The accuracy of depth-from-defocus algorithms is dependent on how well the PSF is modelled and it was desirable to find the PSFs for distances over a reasonable range with sufficient resolution. It was assumed that the test objects would be placed in the range 0 to 300mm from the camera's focus position. As described in Section 3.2.2, the objects must be either all in front or all behind the plane of focus to ensure there is no ambiguity in the depth measurement when only the f-number is changed between images.

Manually moving a camera at small increments and taking duplicate images for averaging purposes due to the significant noise level would be a tedious task. It was decided that an automated approach would allow the tests to be done more quickly with less human error and so a computer-controlled camera moving stage was created. The x-stage was built from an old flatbed scanner. The original electronics were stripped out and a new circuit built that took signals from the computer's parallel port and created the required signals to drive the stepper motor to move the scanning head. An opto-sensor was added to the moving head to detect the starting position. The camera was screwed onto a raised gantry on the old scanner head with an adjustable optical bench post to allow the height and position to be set as required. A Visual Basic program was created that moved the camera at the required increments and interfaced the FireWire camera software to automatically take the images.

The camera travelled over a distance of 312mm in 14,750 steps of the stepper motor and thus had a resolution of 21.2 μm / step. The positional accuracy was tested by resetting the x-stage to its starting position, moving it forward 14,750 steps and then taking an image. The x-stage was reset, the process repeated and another image taken. The two images taken at the first position on the x-stage were then subtracted to produce a difference image, which was then examined for image structure. The difference image only showed noise and thus it was assumed that the positional accuracy was sufficient.

The lightbox was made out of an ABS plastic box with a 25mm wide rectangle removed from the centre of the lid and a thin metal strip was glued to one of the edges to ensure a sharp, straight transition between the light and dark regions. The box was sprayed with grey paint to be a partial scatterer and mounted on a vertical wooden stand with holes so it could be screwed to an optical breadboard and angles marked on for manual orientation of the box. Twenty incandescent bulbs were mounted either side of the slit in order to give an approximately even illumination. The images were taken in a blackened out room to ensure the light levels remained constant over the lengthy image acquisition time. The

bulbs were allowed to settle during a warm-up period before the equipment was used and the brightness fluctuations of the bulbs was not observable as the camera noise dominated.

The lightbox was a piece of legacy equipment originally used by Staunton [57] and it was later found that incandescent bulbs with their low colour temperature were not optimum for colour image processing purposes. This is discussed in Section 8.2.3.

A high frequency strip light that gave a good white colour was employed with a photographic diffuser in front to illuminate the light box cover to ensure that the dark regions were imaged with intensities above the dark level of the camera. Figure 4.5 shows the vertically mounted, rotatable lightbox, the strip lights with a circular diffuser in front and the camera mounted on the gantry on the converted flatbed scanner.

The camera was correctly focused before the focusing mechanism was locked into position and the f-number set. Test images were taken to determine the optimum exposure time to ensure that none of the colour planes were saturating or below their dark level, both of which would contribute undesired non-linear effects. It took 13 hours to collect all of the images for two f-numbers with 18 different angles for distances in 1mm increments with no duplicates.

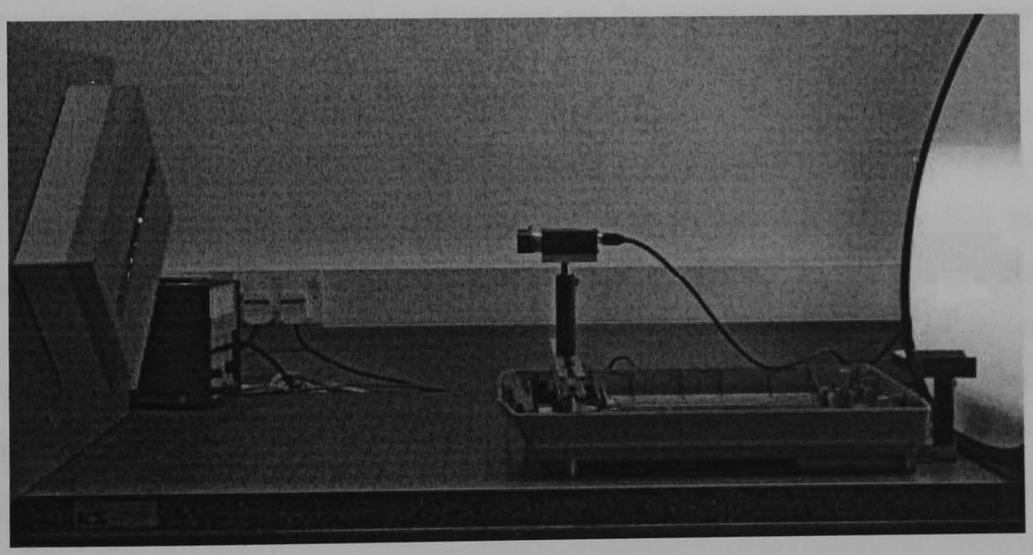


Figure 4.5: The PSF measurement hardware setup

4.5 The PSF Recovery Algorithm

4.5.1 Introduction

Once all of the images had been collected the next stage was to recover the PSF from the ESF for a given image. Section 4.5.2 discusses the demosaicing algorithm used before moving onto the details of the recovery algorithm. Results assuming a Fermi-Dirac fit and pillbox, Gaussian and Generalised Gaussian models are shown in Section 4.5.4. Regularised numerical differentiation was proposed for overcoming the problem with differentiating a noisy signal to find the PSF. Once the all important regularisation term had been determined through simulations the algorithm could then be applied to real data, the results of which are presented in 4.5.5.

4.5.2 The Demosaicing Algorithm

The quantum efficiency of a pixel is dependent on the wavelength of the impinging photons and the properties of the semiconductor in the active area. A 3-CCD camera uses beam splitters and colour filters to produce three versions of the image entering the lens each occupying different spectral bands that then fall on three separate CCDs. The advantage with this approach is high spatial resolution, but at high cost. Another solution for colour imaging is to use one CCD where a colour filter is overlaid during manufacture and the pattern used is called a Colour Filter Array (CFA). One type of CFA uses red, green and blue filters in a mosaic in the ratio $1:2:1$ to mimic the response of the human visual system and another type uses cyan, yellow, green and magenta [24].

In order to produce a colour image a demosaicing algorithm is required, which is a similar process to that required in the human visual system [129] where the colour information comes from three types of cones. The algorithms range from simple one-step procedures to combinations of reconstructions and enhancements [130]. There are many demosaicing algorithms including nearest neighbour replication and interpolation algorithms based on bilinear, bicubic, spline, Laplacian, hue and log hue interpolation methods. It is usually assumed that the pattern of the Colour Filter Array (CFA) does not change throughout the sensor area [131]. Super-resolution can be achieved using a sequence of images where the resolution of the final image is beyond that of the sensor's resolution [131] [130].

A Basler A631fc colour camera with a Bayer filter over its square pixels was used in the research that produces images of size 1388(W)×1038(H) and Basler have a hardware demosaicing algorithm in the camera to produce colour images. Details of the algorithm could not be obtained and since the image was very large to process it was decided to use a simple, known demosaicing algorithm on the raw image and subtract the colour plane offsets to ensure linearity using an algorithm written in MATLAB. The pattern returned by the camera was the repeated form $\begin{pmatrix} R & G \\ G & B \end{pmatrix}$ and a single colour pixel was generated that used the mean of the two green pixels whilst the red and blue components remain unchanged. Each colour image was converted to a greyscale image $I(n, m)$ using the formula

$$I(n, m) = \frac{R(n, m) + G(n, m) + B(n, m)}{3} \tag{4.11}$$

where R, G and B are the red, green and blue colour planes respectively and (n, m) denoted the discrete spatial location. Thus, the spatial resolution of the image was halved in both directions compared to that of the sensor.

In reality the PSF is dependent on the wavelength of light. A well-corrected compound lens uses positive and negative lens elements to reduce chromatic aberration, although it will not be completely eliminated. The higher the spatial resolution of the CCD, the more prominent the effect of chromatic aberration and so down-sampling the colour image by a factor of two in both directions reduces the effect.

4.5.3 The PSF Recovery Algorithm

The colour images are demosaiced and converted to monochrome images as described in the previous section and Figure 4.6 shows an example image from the Sigma 24mm lens with an f-number of f/2.8 when the distance between the lens and the lightbox was 0.725m.

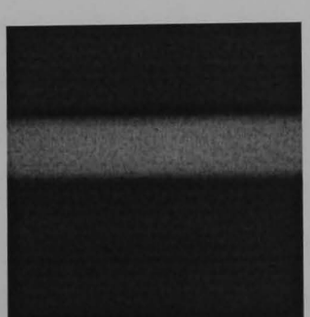


Figure 4.6: An example of an image used to recover the 1D PSF

Staunton [57] used a 7×7 Integrated Directional Derivative (IDD) edge detector to locate the edges in a light box image, i.e. the transition from the light to dark, but it was found that for defocused edges the detector failed. A Canny edge detector was employed

to solve the problem as it worked well for both focused and defocused edges and importantly it accurately located the centre of the edge even with significant blurring. The parameters of the Canny edge detector were tuned emphirically so that no false edges were found and the optimum standard deviation of the Gaussian filter was $\sigma = 3$ and the low and high thresholds were $T_1 = 0.3$ and $T_2 = 0.7$ respectively.

For a range of distances at the furthest extent from the light box both edges of the slit were visible, only one of which was the knife edge. An algorithm was written to leave only the required edge in the edge detected image. The detected edge was then used as the centre of a rectangular window that was applied to the greyscale lightbox image. The width of the rectangle perpendicular to the edge was 51 pixels so that an ESF has 51 samples, where an ESF is defined as the intensities perpendicular to the edge, which correspond to a column in the windowed image. If the angle of the edge was not a multiple of 90 degrees then the required pixel positions did not line up with the sampling grid and so nearest neighbour interpolation was employed. It was important that the edge was not a multiple of 90 degrees so that over-sampling occurred.

Figure 4.7: An example of the windowed image

A single Edge Spread Function (ESF) was formed from samples perpendicular to the edge and the windowing algorithm was improved to extract the maximum number of complete ESFs. Experimentally it was found that the image was brightest near the centre of the lightbox and the ESFs were normalised to remove non-uniform illumination effects along the direction of the edge. The edge detection only approximately located the centre of the brightness transition and so the 50% brightness points had to be aligned. Staunton's [57] original algorithm used a linear fit of the central intensity values in the ESF, but it was found inadequate for defocused edges and a cubic fit was employed instead. The effect of aligning the centres of the edges meant that the sample points were displaced relative to each other. The super-resolution edge was created by averaging the pixel intensities within pixel bins to give a ten times resolution improvement.

Having obtained the mean ESF for a given distance, f-number and lightbox angle it was necessary to find the PSF and the different methods that were examined are:

- Five-point numerical differentiation
- Regularised numerical differentiation using Chartrand's algorithm
- Regularised numerical differentiation using Chartrand's algorithm followed by a fit of the resulting PSF to a Generalised Gaussian function

- Fitting the ESF to a sum of Fermi-Dirac functions as described by Tzannes *et al.* [56]
- Fitting the ESF to a defocused step assuming even illumination and a Gaussian PSF
- Fitting the ESF to a defocused step where the illumination is assumed to have a linear dependence on position and a Gaussian PSF
- Fitting the ESF to a defocused step assuming even illumination and a Generalised Gaussian PSF
- Fitting the ESF to a defocused step where the illumination is assumed to have a linear dependence on position and a Generalised Gaussian PSF

The mean ESF was fitted to a sum of Fermi-Dirac functions and ESFs assuming pill-box, Gaussian and Generalised Gaussian PSFs, examples of which are presented in the next section. A regularised numerical differentiation algorithm was discussed in Section 3.4.7 and the results are shown in Section 4.5.5.

4.5.4 Specific 1D Results

In this section results for the Sigma 24mm lens fitted to the Basler A631fc colour camera are presented when the lightbox was 0.725m from the camera and the lightbox angle was approximately 0 degrees, but not exactly to ensure super-resolution could be achieved. The PSFs have been normalised to be in the range [0, 1] to highlight the differences in the shape.

The results from the five-point numerical differentiation in Figure 4.8 show that although the ESF looks fairly smooth, the noise is swamping the underlying PSF, thus making this approach unusable without further processing.

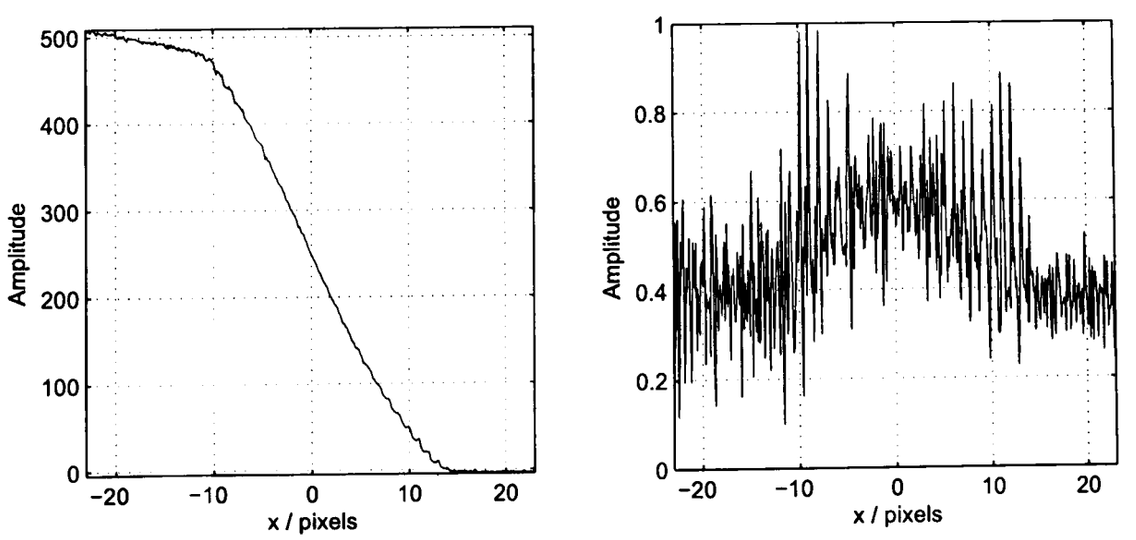


Figure 4.8: Five-point numerical differentiation results for f/2.8, z=0.725m, angle=0 degrees with ESF shown on the left and the PSF on the right

It was believed that the noise on the ESF was due to camera noise and not fluctuations in the intensities of the bulbs. The analysis in Section 3.4.7 showed that for a smooth signal and uncorrelated additive noise that the gradient of the noise is greater than that due to the signal. Thus, the derivative of the noise swamped the derivative of the ESF, the latter of which was the required PSF.

Tanzes *et al.* [56] fitted their ESF to a sum of Fermi-Dirac functions and Figure 4.9 shows the result. The ESF has a very good fit, however the PSF neither has symmetry or a single peak, two properties expected of a physical PSF.

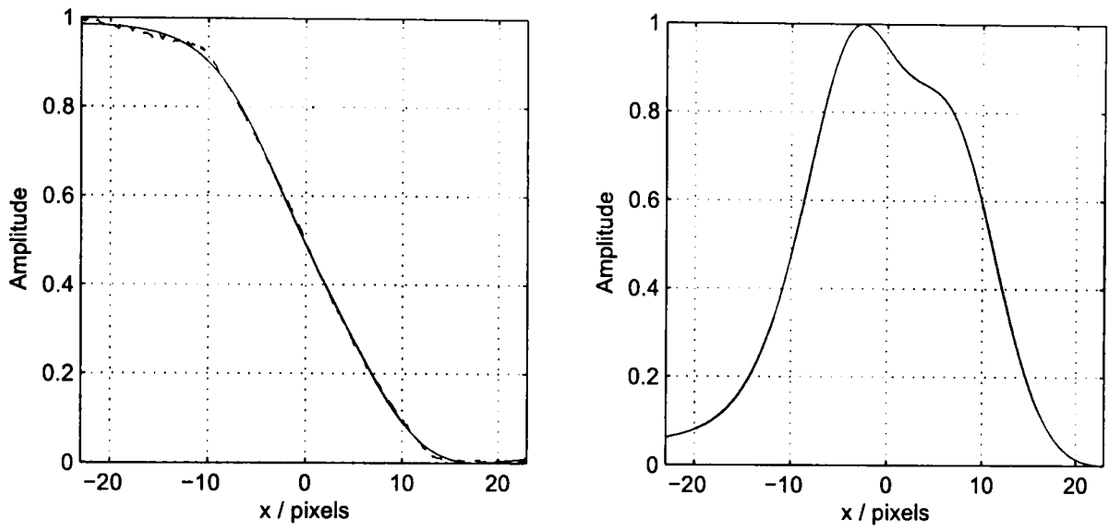


Figure 4.9: The actual ESF (dashed line) and Fermi-Dirac fitted ESF (solid line) results for f/2.8, z=0.725m, angle=0 degrees (MSE = 4.00×10^{-5})

The results of using the novel PSF shape of the Generalised Gaussian are shown in Figure 4.10 and Figure 4.11. The shape of the Generalised Gaussian is naturally dependent on whether the non-uniform illumination is taken into account. The MSE assuming a Generalised Gaussian and the non-uniform illumination is the lowest and the PSF is of an acceptable shape. Thus, better results have been achieved by using an improved illumination model.

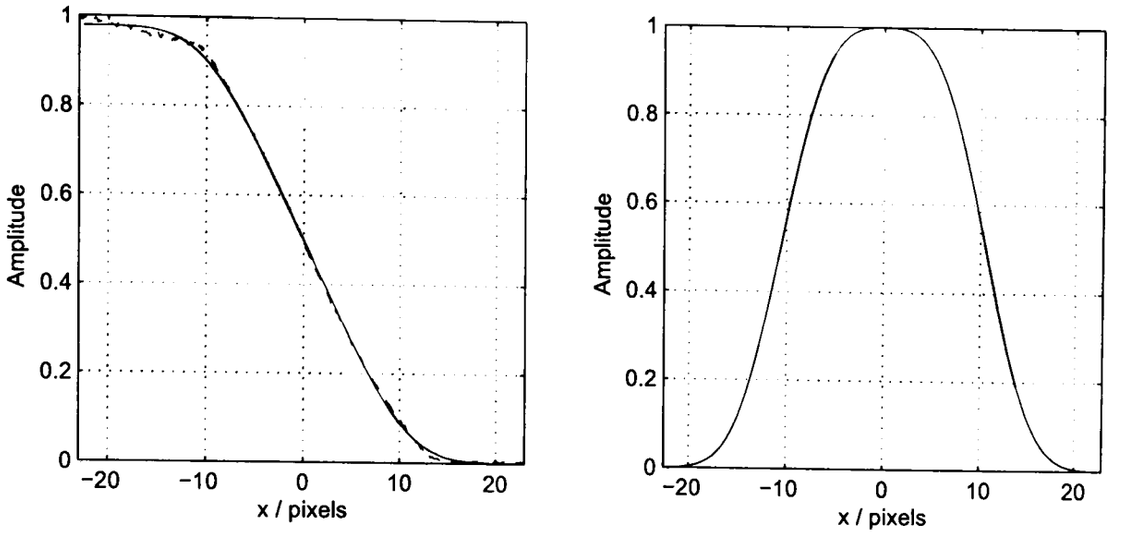


Figure 4.10: Actual ESF (dashed line) and Generalised Gaussian without illumination correction fitted ESF (solid line) results for f/2.8, z=0.725m, angle=0 degrees (MSE = 5.67×10^{-5})

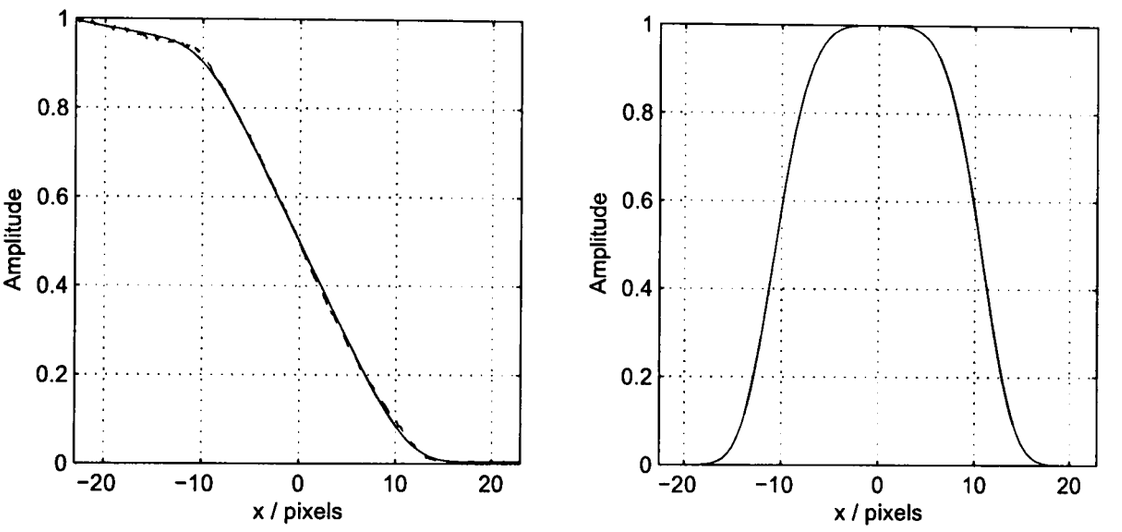


Figure 4.11: Actual ESF (dashed line) and Generalised Gaussian with illumination correction fitted ESF (solid line) results for f/2.8, z=0.725m, angle=0 degrees (MSE = 3.63×10^{-5})

The Gaussian PSF without taking the illumination into account has a good fit (see Figure 4.12), but it is clear that the gradient in the illumination has resulted in a smaller σ than that obtained when taking into account the non-uniform brightness, shown in Figure 4.13. Assuming uniform illumination, the only parameters in the fit of the actual ESF to a model ESF are the standard deviation σ, the mean x_0 and the upper and lower intensities of the step m_1 and m_2. With non-uniform illumination taken into account two more parameters are optimised, which are the gradients of the step, c_1 and c_2. Thus, with uniform illumination $c_1 = c_2 = 0$ and with a non-uniform illumination model they are optimised. The fitting algorithm blindly finds the optimum parameters to reduce the error between the actual ESF and the fitted ESF, therefore it cannot be expected that the standard deviations of the Gaussians will be identical regardless of how the illumination is

taken into account. The MSE between the fitted ESF and the actual ESF reduced from 1.08×10^{-4} to 8.78×10^{-5} by taking into account the non-uniform illumination, which is a reduction of 19%.

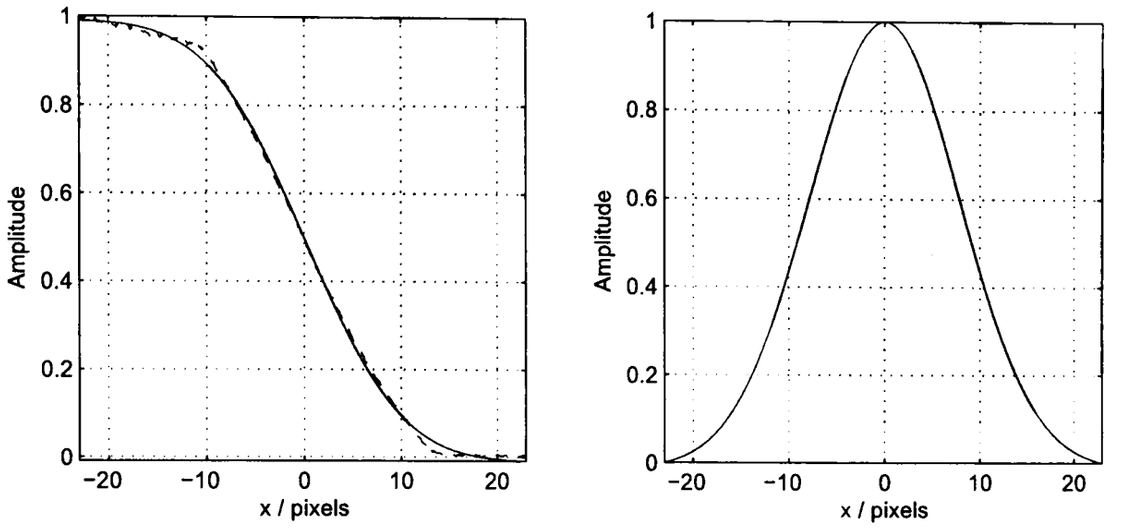


Figure 4.12: Gaussian without illumination correction results for f/2.8, z=0.725m, angle=0 degrees (MSE = 1.08×10^{-4})

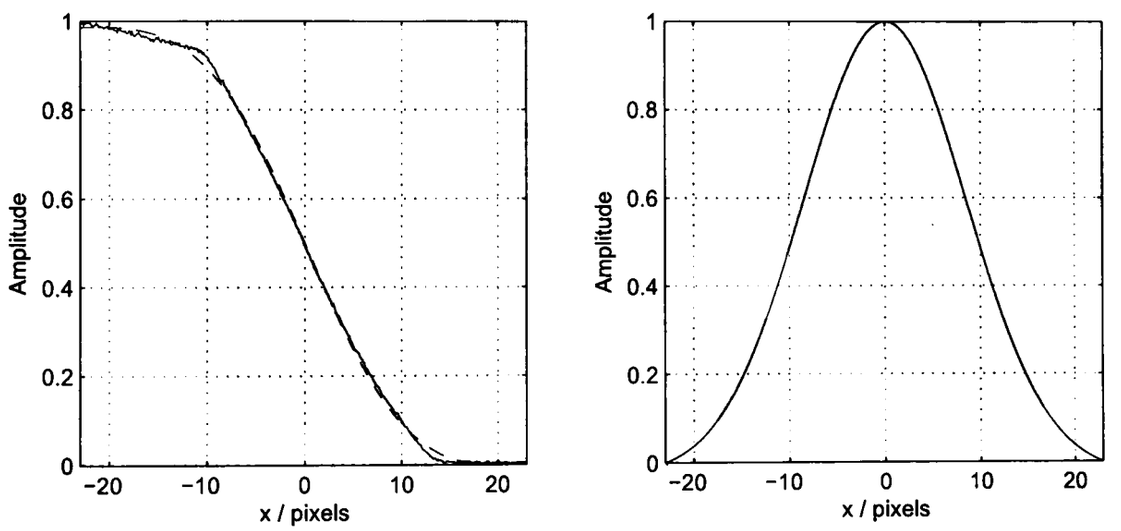


Figure 4.13: Gaussian with illumination correction results for f/2.8, z=0.725m, angle=0 degrees (MSE = 8.78×10^{-5})

In the results presented here the camera is very defocused and a good fit assuming a pillbox PSF is shown in Figure 4.14 and Figure 4.15, however, the MSE is greater than all the other methods.

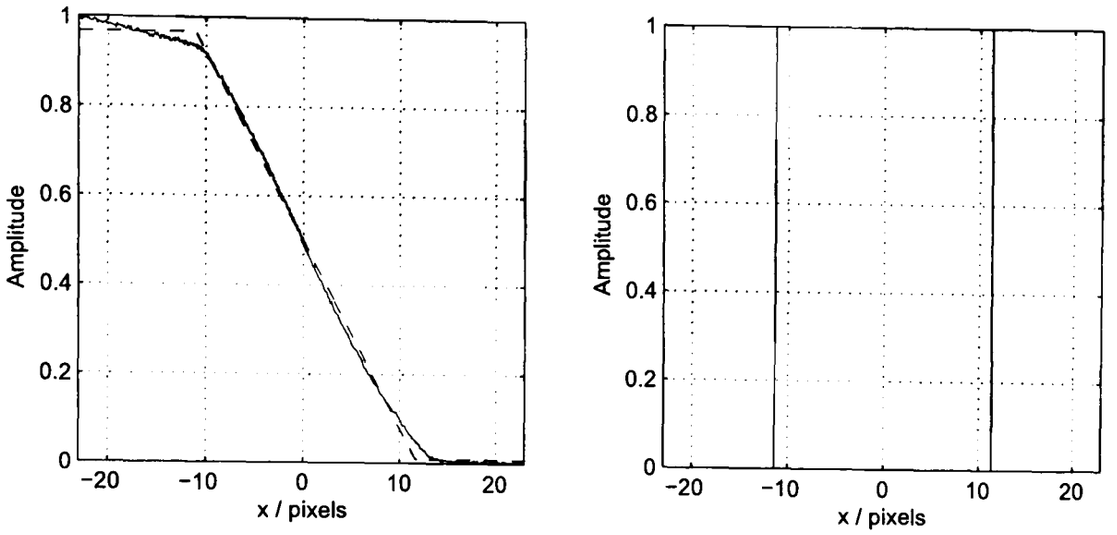


Figure 4.14: Pillbox without illumination correction results for f/2.8, $z = 0.725$m, angle = 0 degrees (MSE = 2.18×10^{-4})

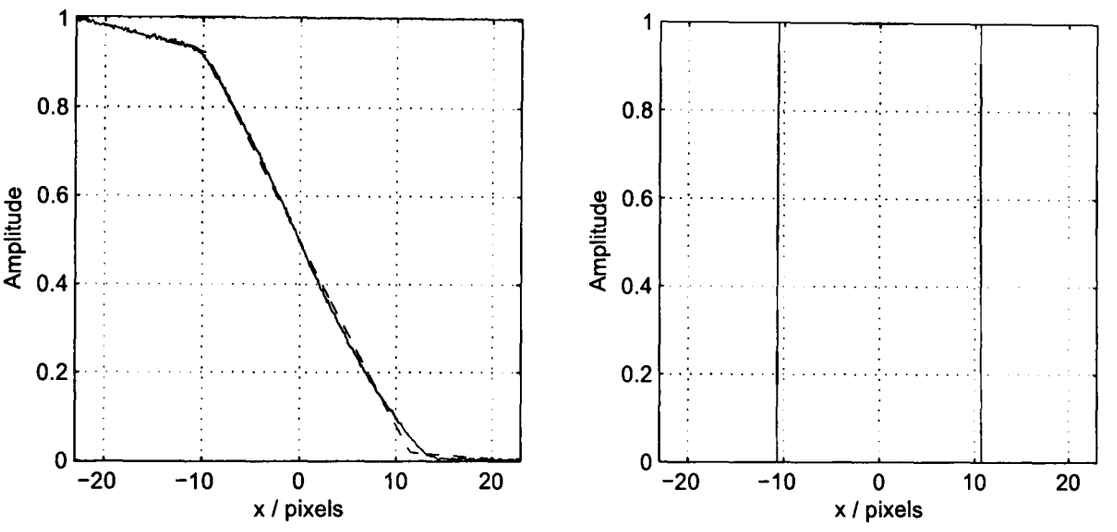


Figure 4.15: Pillbox with illumination correction results for f/2.8, z=0.725m, angle=0 degrees (MSE = 9.39×10^{-5})

4.5.5 Regularised Numerical Differentiation

In order to determine the optimum regularisation parameter α for PSF measurement a series of simulations were performed. Pillbox and Gaussian PSFs were used to defocus blur an ideal step, noise was added and then the ESF differentiated using Chartrand's algorithm [126]. The mean square error (MSE) was employed as a distance measure between the actual PSF and the result of the numerical differentiation. The figures below show plots of the MSE as a function of α for pillbox and Gaussian PSFs with a signal-to-noise ratio (SNR) of 30 dB. From the experiment it was determined that the value α = 100 served both PSFs well for the range of SNRs and thus it was employed in tests on real ESFs.

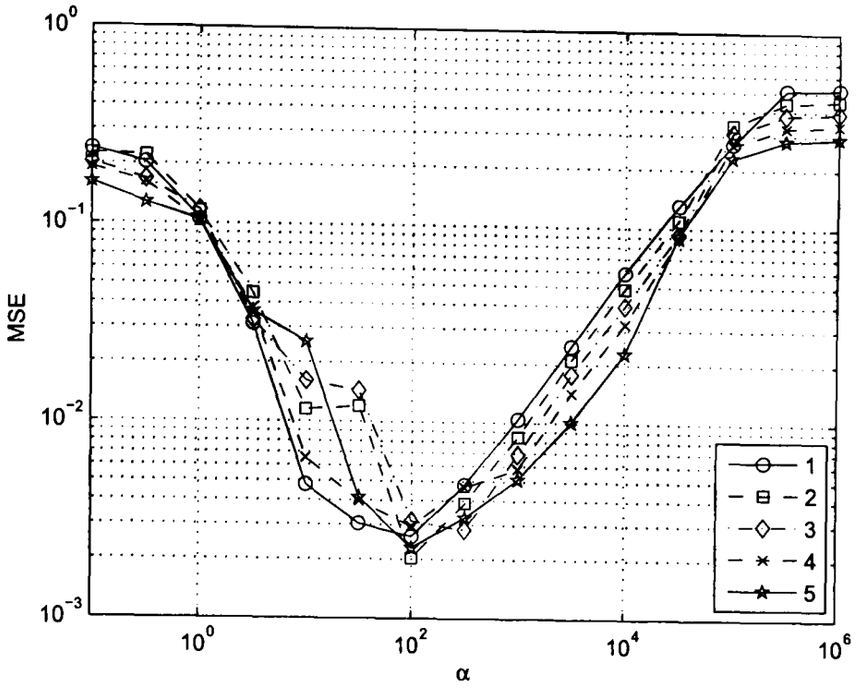


Figure 4.16: The MSE between the recovered PSF and the actual Gaussian PSF for standard deviations of 1 to 5 pixels

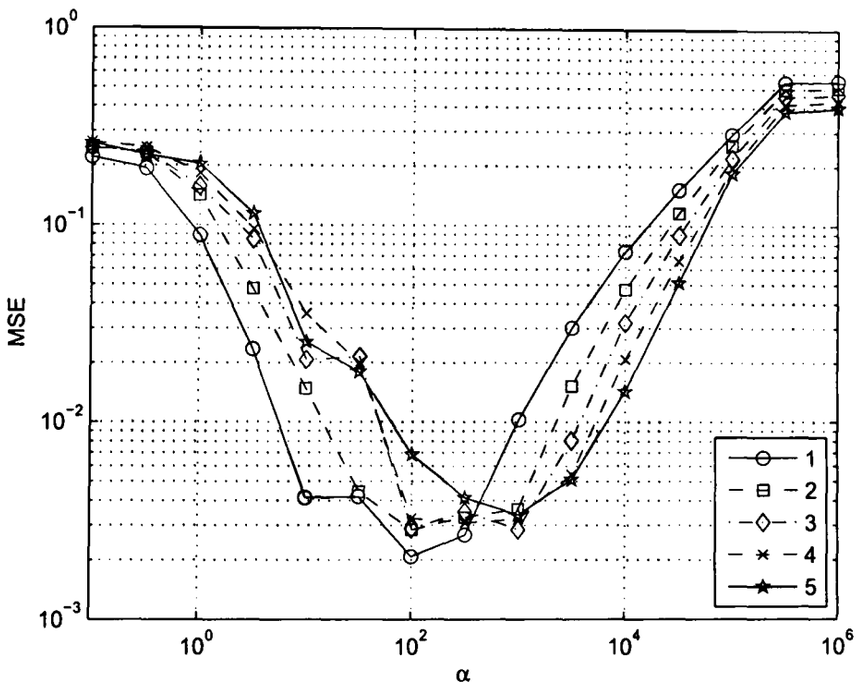


Figure 4.17: The MSE between the recovered PSF and the actual Pillbox PSF for blur circle radii of 1 to 5 pixels

The ESF shown in 4.5.4 was differentiated using Chartrand's regularised numerical differentiation (RND) algorithm [126] and Figure 4.18 shows the PSFs when α = 10, 100, 1000 and it can be seen that the function gets smoother as α increases, as expected. Note that the linear brightness change on the upper step level has produced a constant value in the derivative.

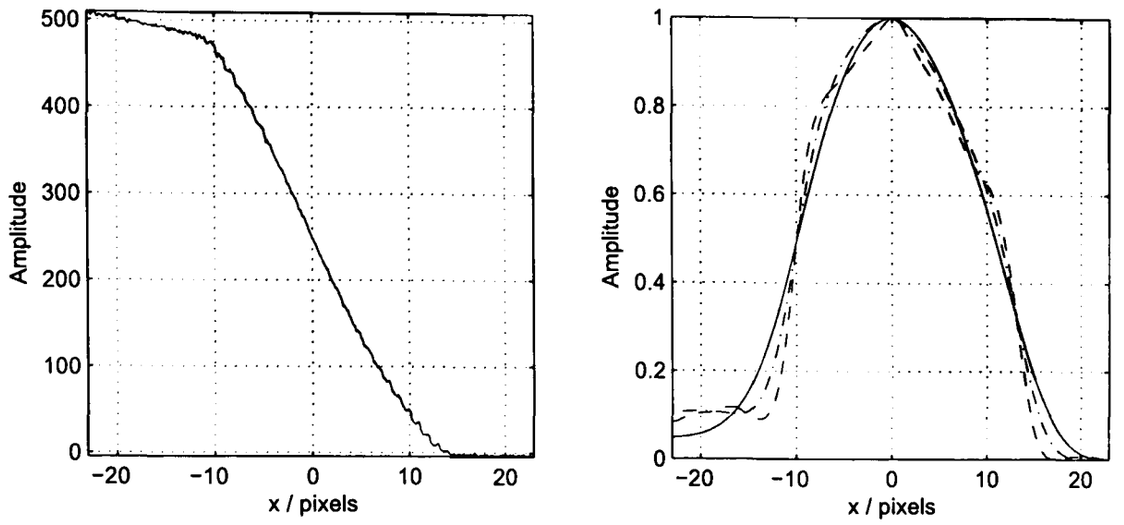


Figure 4.18: ESF (left) and regularised numerical differentation results (right) for $\alpha = 10$ (dashed), $\alpha = 100$ (dash-dot) and $\alpha = 1000$ (solid)

A Generalised Gaussian was fitted to the resultant PSFs when four different depth positions were tested using $\alpha = 1000$, the results of which are displayed in Figure 4.19 and Figure 4.20.

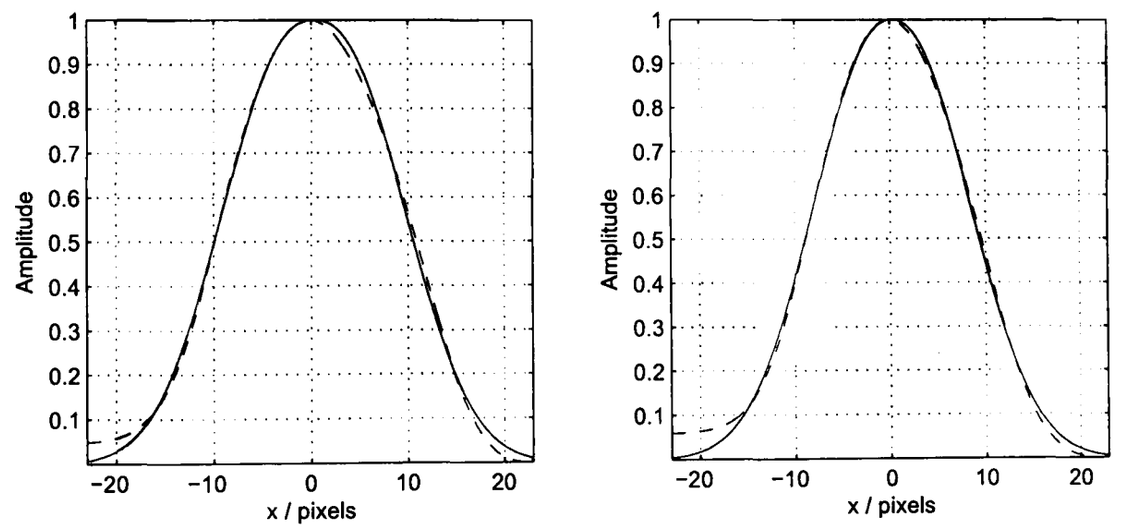


Figure 4.19: The regularised numerical differentiation PSF (dashed) and the fitted Generalised Gaussian (solid) for depths of 0.725m (left) and 0.647m (right)

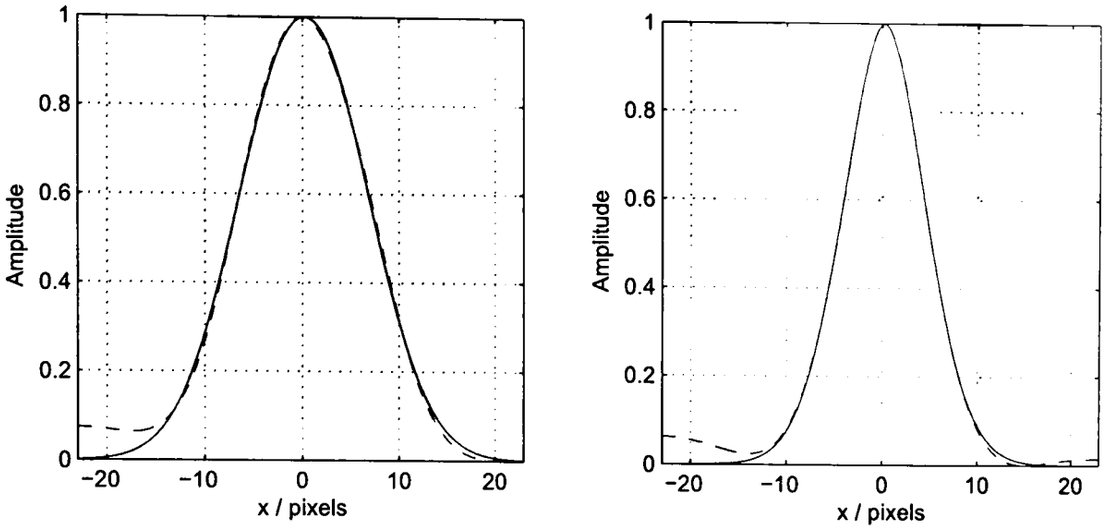


Figure 4.20: The regularised numerical differentiation PSF (dashed) and the fitted Generalised Gaussian (solid) for depths of 0.569m (left) and 0.414m (right)

The results of the fit are summarised in Table 4.2 and it will shown in Section 4.7.4 that the parameters of the Generalised Gaussian fitted to the result of the regularised numerical differentiation result do not appear favourable compared to using a Generalised Gaussian-based ESF fit from the beginning.

Table 4.2. Results from fitting a Generalised Gaussian to the RND PSF

Depth / m	MSE	Power, p	Standard deviation, σ
0.414	0.209	1.84	4.37
0.569	0.283	2.09	6.43
0.647	0.185	2.23	7.52
0.725	0.148	2.40	8.29

As discussed in Section 3.4.6 the standard deviation of the Generalised Gaussian is a measure of the spatial spread and the power specifies its shape. As p decreases from 2 towards 0 the Generalised Gaussian becomes more pointed. At $p = 2$ it simplifies to a Gaussian and as p increases towards infinity the function approximates a pillbox with decreasing error.

4.6 Results for the 16mm Video Lens

A 16mm Basler video lens was tested for three different apertures (f/1.4, f/2 and f/4) and the resulting standard deviation σ of the PSFs recovered assuming a Gaussian PSF are shown in Figure 4.21. The focus position of the camera was not altered during the experiments, but clearly the point of best focus shown by a minimum in σ changes with f-number. The focus distances of f/1.4, f/2 and f/4 apertures are 0.464m, 0.503m and 0.568m respectively and this effect can be attributed to the presence of spherical aberration. Spherical aberration is caused by a lens that focuses the marginal rays closer to the lens than the paraxial rays [132] and thus the focal length is dependent on the aperture for non-paraxial rays [2].

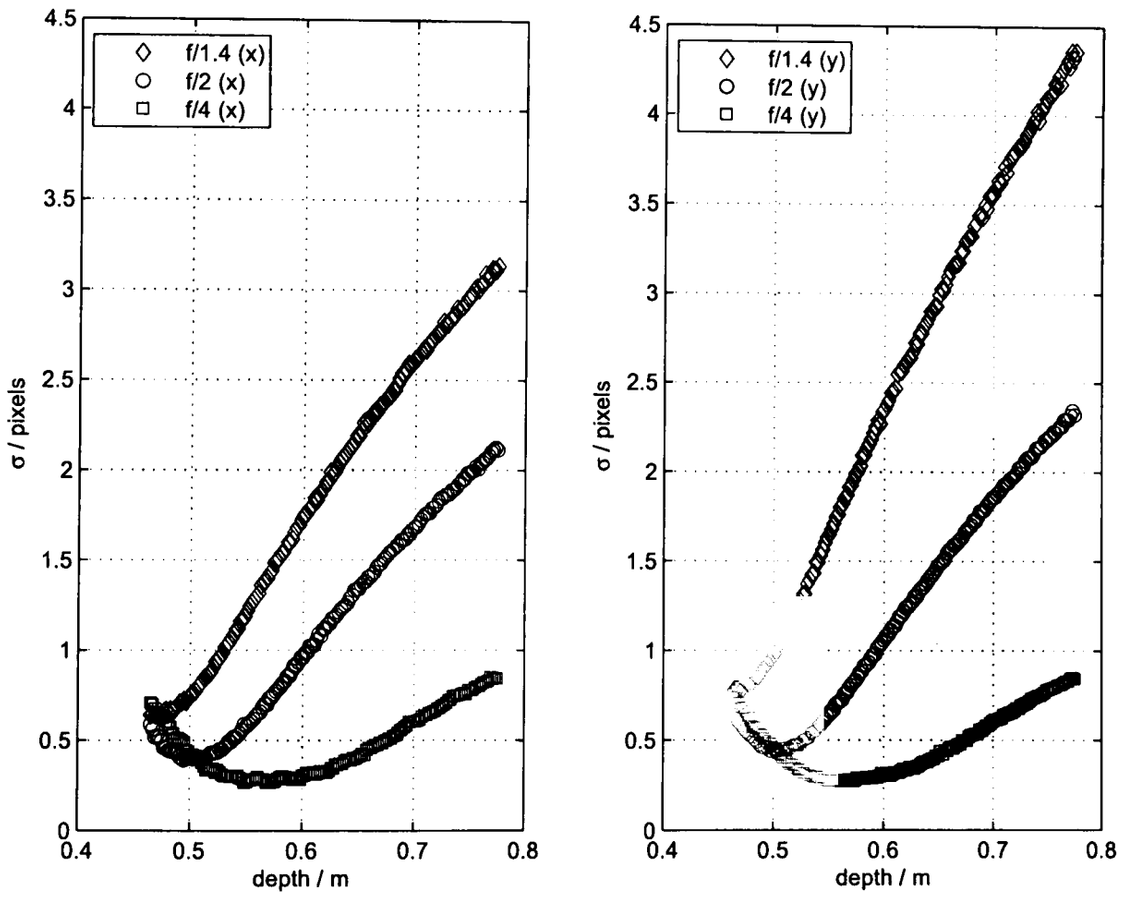


Figure 4.21: Gaussian PSF results for the video lens for the horizontal (left) and vertical (right) directions

For a circular aperture with square pixels it is expected that the PSF would possess circular symmetry, especially with a lot of defocus. The fact that the circular symmetry is not present for wider apertures suggests that other aberrations could be present and in particular coma and astigmatism would cause a non-circularly symmetric PSF [100]. A further problem was that the edges of the image were appreciably defocused while the centre region was in focus, thus clearly the PSF is not space-invariant.

4.7 Results from the 24mm Sigma Photographic Lens

4.7.1 Introduction

Due to the problems with the video lens, a high element count, good quality 24mm photographic lens was used in the subsequent tests. A mount was sourced to allow an SLR lens to fit into the C-mount of the Basler camera. The distance between the back of the lens and the CCD was found to be the same as would be used in an SLR camera between the lens and film plane. Thus, no aberrations were expected as a result of the C-mount. The next section shows the MSE of fitting the actual ESF to the theoretical models and then Section 4.7.3 and 4.7.4 show complete results for the Gaussian and Generalised Gaussian PSF models respectively. Finally, some 2D results are shown to illustrate the complete form of the PSF.

4.7.2 Edge Spread Function Fitting Experiments

Depth-from-defocus requires accurate knowledge of the PSF of the lens for given settings. The ESFs from the lightbox images were fitted to various different functions for a range of distances. The results below in Tables 4.3 to 4.5 show the mean square error of the fit as an average for all angles tested, which were -80 to $+90$ degrees in 10 degree intervals.

The results in Tables 4.3 to 4.5 show that the error assuming a pillbox PSF decreases for increasing defocusing, which was expected from the theoretical diffraction-based optics approach in Section 3.3.2. The mean square errors of the fits using Generalised Gaussian, Gaussian and pillbox models are lower when taking into account the non-uniform illumination compared to assuming uniform illumination. In particular, for the Generalised Gaussian fit at a depth of 0.414m with an aperture of f/5.6, the MSE was halved by incorporating the improved illumination model.

Table 4.3. MSE results for f/2.8 as a function of the depth of the light box (to 3 s.f.)

	Mean Square Error (MSE) / 10^{-3}				
Method	0.414m	0.491m	0.569m	0.647m	0.725m
Fermi-Dirac	25.2	29.2	34.3	28.3	26.1
Generalised Gaussian without I.C.	10.3	7.37	9.03	5.58	6.45
Generalised Gaussian with I.C.	7.91	5.92	7.95	4.99	6.01
Gaussian without I.C.	64.6	51.1	64.9	68.2	70.2
Gaussian with I.C.	47.6	43.4	55.0	51.1	48.5
Pillbox without I.C.	130	90.9	90.7	86.0	85.5
Pillbox with I.C.	102	70.3	72.4	70.8	68.3

Table 4.4. MSE results for f/4 as a function of the depth of the light box (to 3 s.f.)

	Mean Square Error (MSE) / 10^{-3}				
Method	0.414m	0.491m	0.569m	0.647m	0.725m
Fermi-Dirac	25.5	26.6	23.3	24.4	30.7
Generalised Gaussian without I.C.	7.60	6.81	5.65	5.34	4.91
Generalised Gaussian with I.C.	5.31	5.06	4.25	4.58	4.18
Gaussian without I.C.	63.3	39.3	44.4	49.6	58.4
Gaussian with I.C.	44.2	30.9	36.7	39.9	44.4
Pillbox without I.C.	138	94.1	92.1	91.8	96.8
Pillbox with I.C.	107	69.8	67.2	68.5	86.2

Table 4.5. MSE results for f/5.6 as a function of the depth of the light box (to 3 s.f.)

Method	Mean Square Error (MSE) / 10^{-3}				
	0.414m	0.491m	0.569m	0.647m	0.725m
Fermi-Dirac	15.8	25.3	27.6	29.5	28.4
Generalised Gaussian without I.C.	7.17	7.89	5.99	7.03	6.84
Generalised Gaussian with I.C.	3.08	4.67	3.20	4.12	4.39
Gaussian without I.C.	73.7	45.6	43.5	54.9	59.3
Gaussian with I.C.	42.9	29.7	31.8	43.3	48.8
Pillbox without I.C.	132	87.7	78.6	85.0	84.5
Pillbox with I.C.	91.2	55.7	47.4	51.1	52.8

Table 4.6. Mean MSE results for all three apertures from best to worst

Method	Average MSE / 10^{-3}
Generalised Gaussian with Illumination Correction	5.04
Generalised Gaussian without Illumination Correction	6.93
Sum of three Fermi-Dirac functions	26.7
Gaussian with Illumination Correction	42.5
Gaussian without Illumination Correction	56.7
Pillbox with Illumination Correction	72.0
Pillbox without Illumination Correction	97.6

The summarised results in Table 4.6 show that the Generalised Gaussian with illumination correction has resulted in the lowest MSE, thus giving the best fit to the data. The geometrical optics derived pillbox model produced the worst results with a MSE about 14 times greater than that of the Generalised Gaussian. The MSE of the Gaussian fell almost half way between the Generalised Gaussian and the pillbox and the MSE is 8 times worse than that due to the Generalised Gaussian.

The incorporation of the non-uniform illumination into the model has decreased the MSE using the Generalised Gaussian, Gaussian and pillbox models by 27.3%, 25.0% and 26.2% respectively. Thus it can be concluded that the non-uniform illumination consideration is very important when recovering the PSF of a defocused lens.

4.7.3 Results assuming a Gaussian PSF

Images of the lightbox were obtained in 1mm increments over a 30cm range for angles of −80 to +90 degrees in 10 degree increments. Each image gives a single mean ESF and that ESF was fitted assuming a Gaussian PSF, as derived in Section 3.4.5. The PSF was found to be very nearly circularly symmetric and so Figure 4.22 shows the standard deviation of the Gaussian as a function of distance for three different f-numbers under test. The data appears to be very smooth, except close to the maximum distance tested. The x and y-direction data has been shown in separate figures as the data overlaps almost exactly.

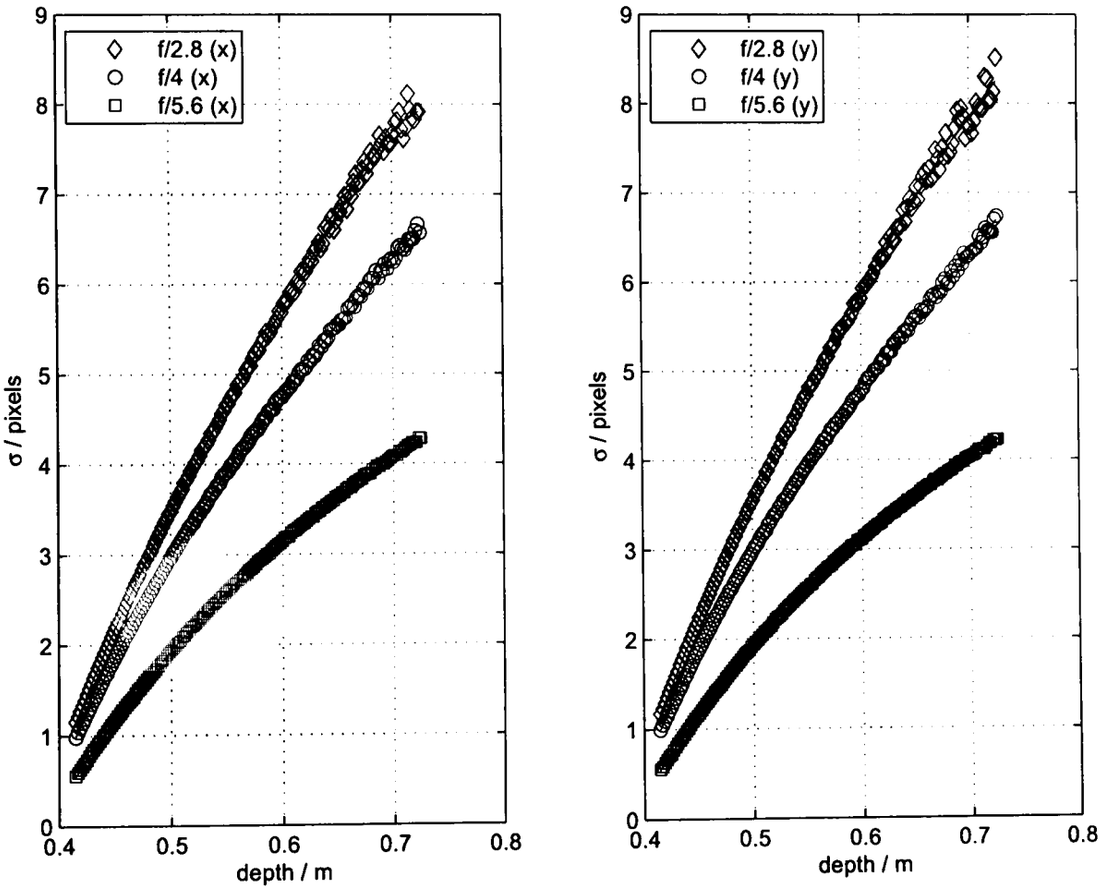


Figure 4.22: Results from fitting a Gaussian PSF in the (left) x-direction and (right) y-direction

The results presented in Section 4.6 for the video lens showed that it suffered from spherical aberration, which caused the focus position to change with f-number, and the PSF was definitely not circularly symmetric. The results in Figure 4.22 do not show any spherical aberration problems in contrast and the PSF is circularly symmetric.

In order to show how the defocusing affects the PSF Figure 4.23 shows the PSFs for depths of 0.414m, 0.491m, 0.569m, 0.647m and 0.725m.

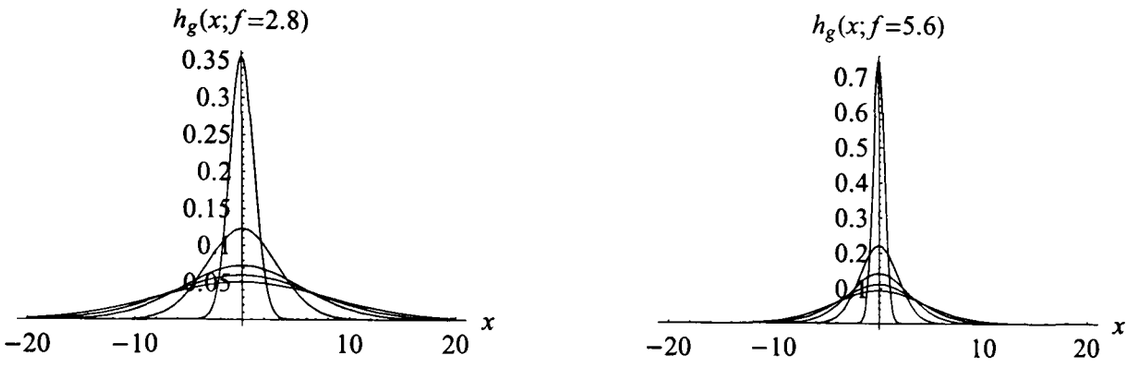


Figure 4.23: PSFs for the Gaussian fit when the lens was progressively defocused for f/2.8 (left) and f/5.6 (right)

The diffraction model was presented in Section 3.3.3, a Gaussian was fitted to the model and the parameters used for the diffraction model were the same as set for the camera. The actual results and the diffraction model are presented in Figure 4.24 and it can be seen that the shapes of the expected curves are similar to that recovered in practice, but the alignment is not very good. The diffraction model neglects aberrations and sampling and the camera parameters are only known approximately.

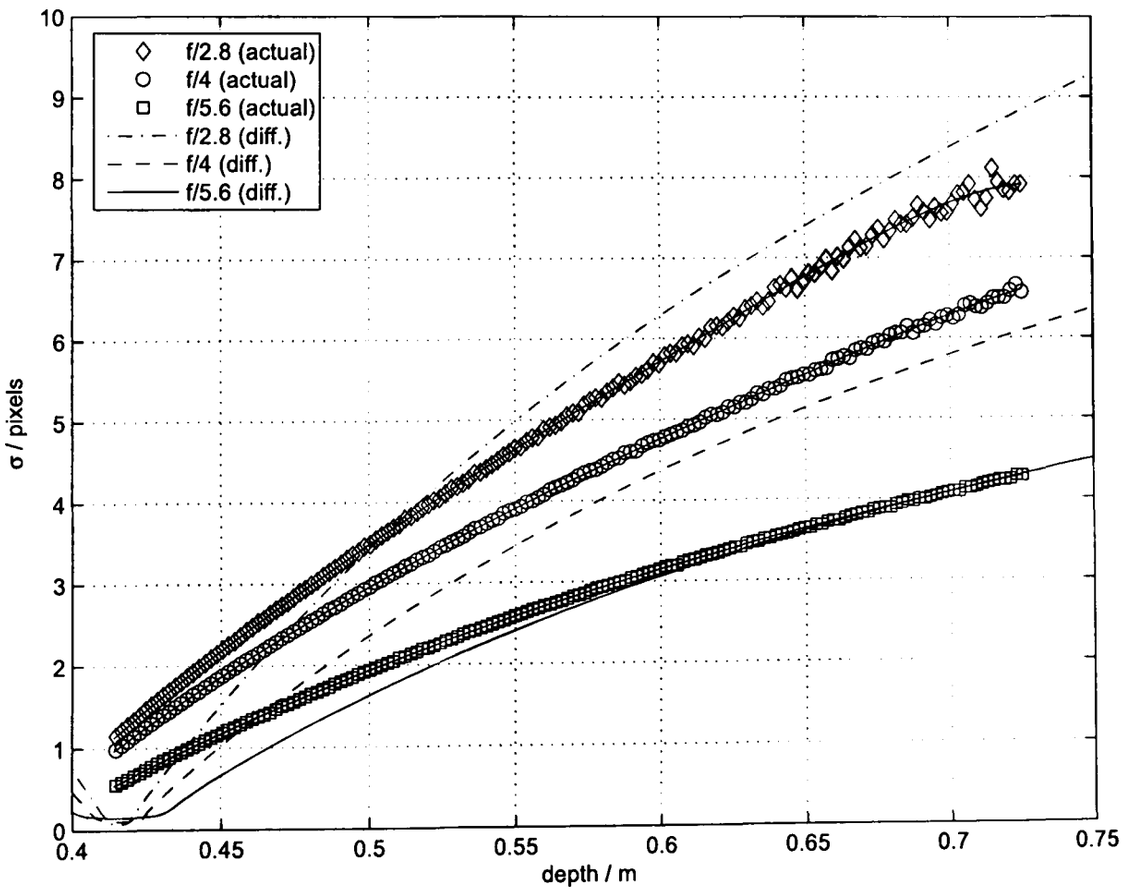


Figure 4.24: Actual (points) and diffraction-based model (lines) for the Sigma 24mm lens

4.7.4 Results assuming a Generalised Gaussian PSF

The Generalised Gaussian PSF has two parameters: the standard deviation σ; and the power p. In Figures 4.25 to 4.27 the standard deviations and powers of the Generalised Gaussians for three different f-numbers are shown. The standard deviation are very smooth, as with the Gaussian fit, however the powers as a function of depth appear much more noisy.

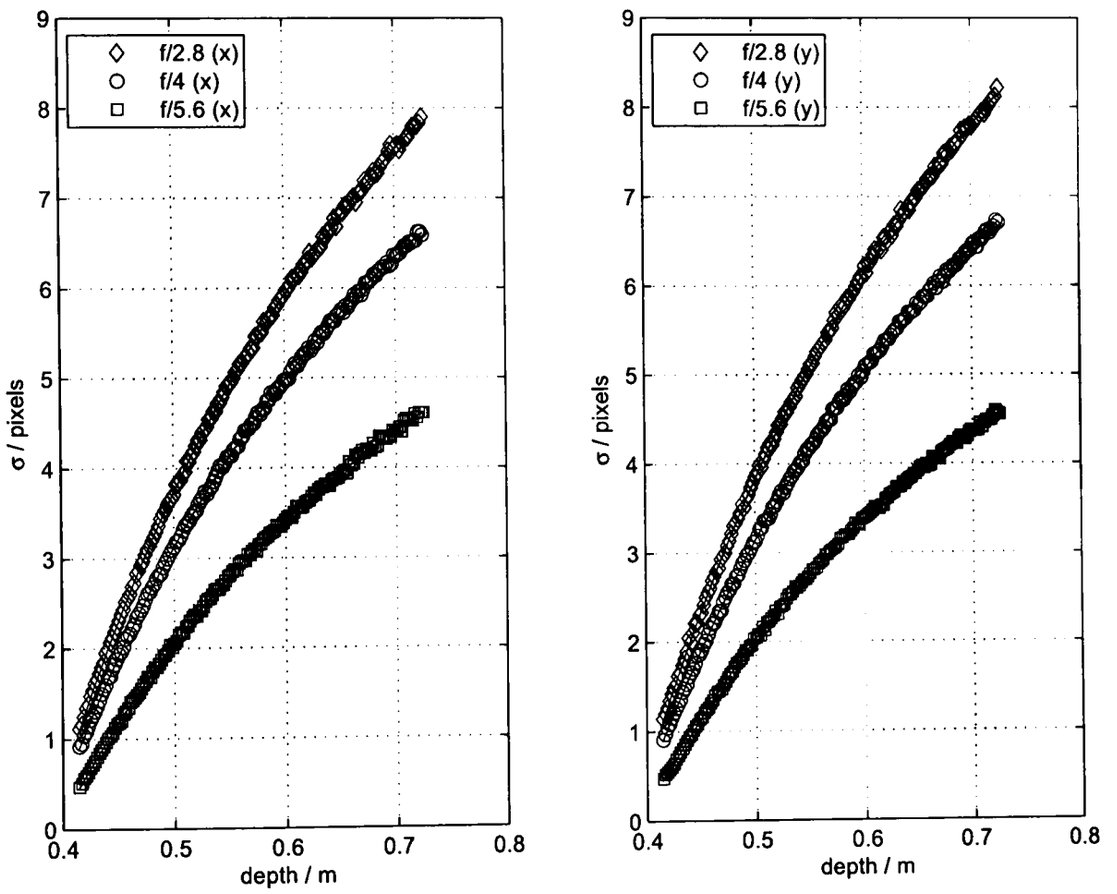


Figure 4.25: The standard deviation of the Generalised Gaussian for x- (left) and y-directions (right)

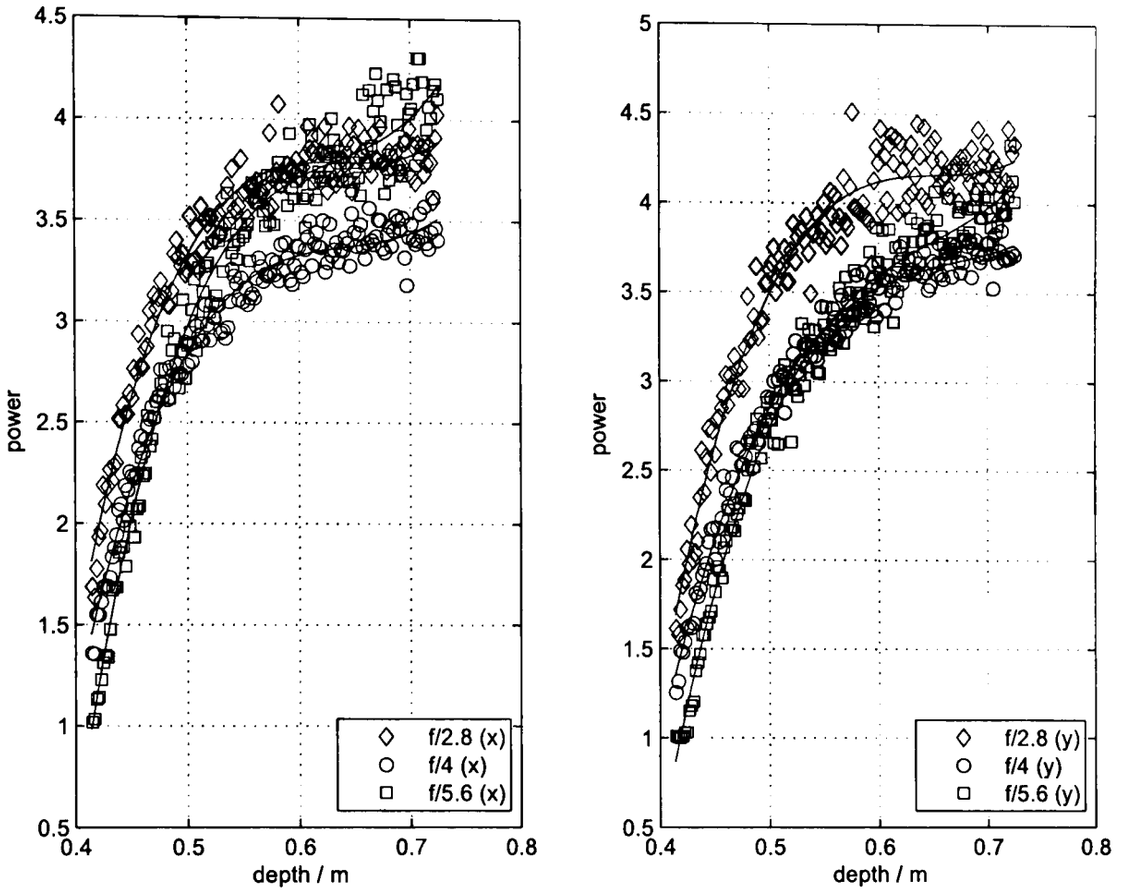


Figure 4.26: The power of the Generalised Gaussian for x- (left) and y-directions (right)

In Figure 4.27 the shape of the power of the Generalised Gaussian versus depth is shown accurately using lines and the symbols are purely for identification purposes as there was so much data. Each set of data was fitted to a 6^{th} order polynomial for smoothing purposes.

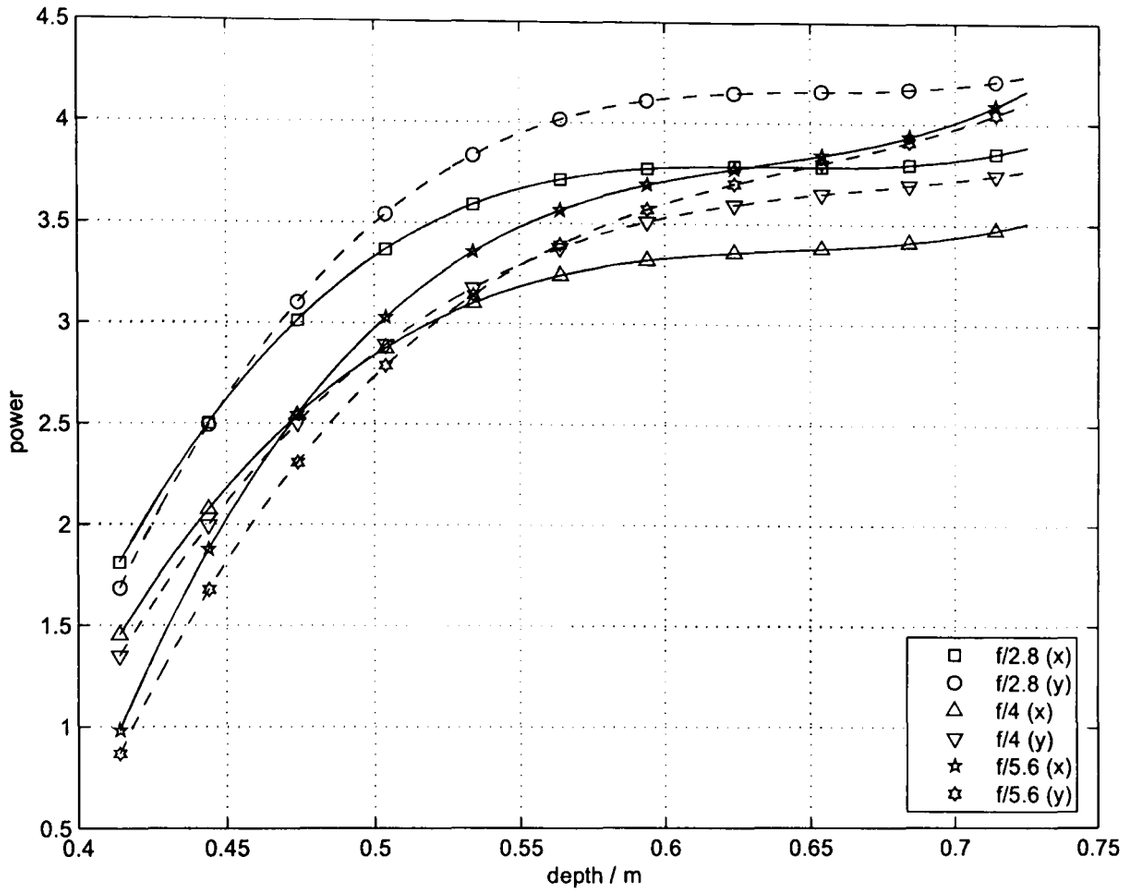


Figure 4.27: The power of the Generalised Gaussian for x- (left) and y-directions (right) where only the fitted data is presented

In order to show how the defocusing affects the PSF, the PSFs for depths of 0.414m, 0.491m, 0.569m, 0.647m and 0.725m are shown in Figure 4.28.

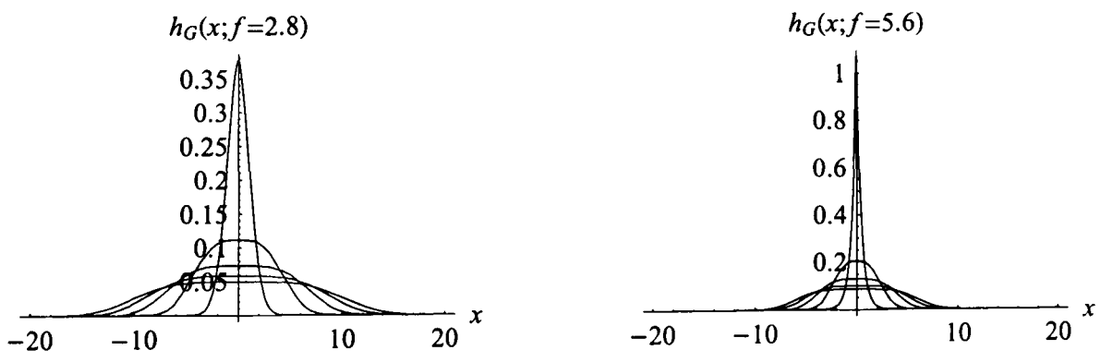


Figure 4.28: Generalised Gaussian fit for f/2.8 (left) and f/5.6 (right) for a progressively defocused lens

4.7.5 Two-dimensional PSFs

The results thus far have focused on 1D PSFs, which are sections through the complete 2D PSF. Now the complete PSFs are presented assuming a pillbox, Gaussian and Generalised Gaussian PSF models for two depths, namely 0.725m and 0.414m, corresponding to the furthest and closest positions tested. The non-uniform illumination improvement was used. Figures 4.29 to 4.34 show the PSFs for a particular distance between the camera and the lightbox (denoted z in the figure labels) for an aperture of f/2.8.

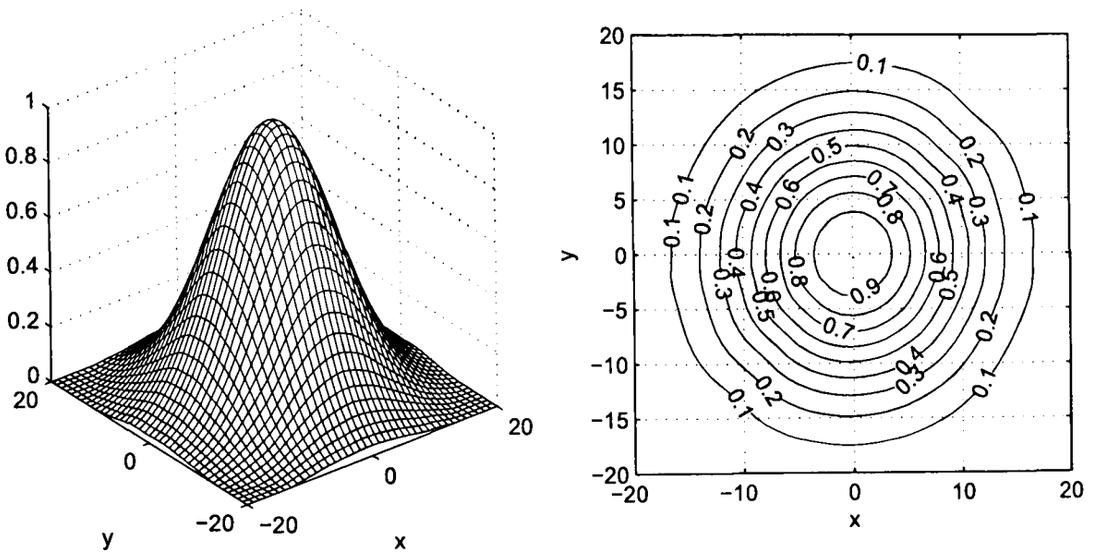


Figure 4.29: 2D PSF assuming a Gaussian model for $z = 0.725$ m and f/2.8 where x and y are in pixels

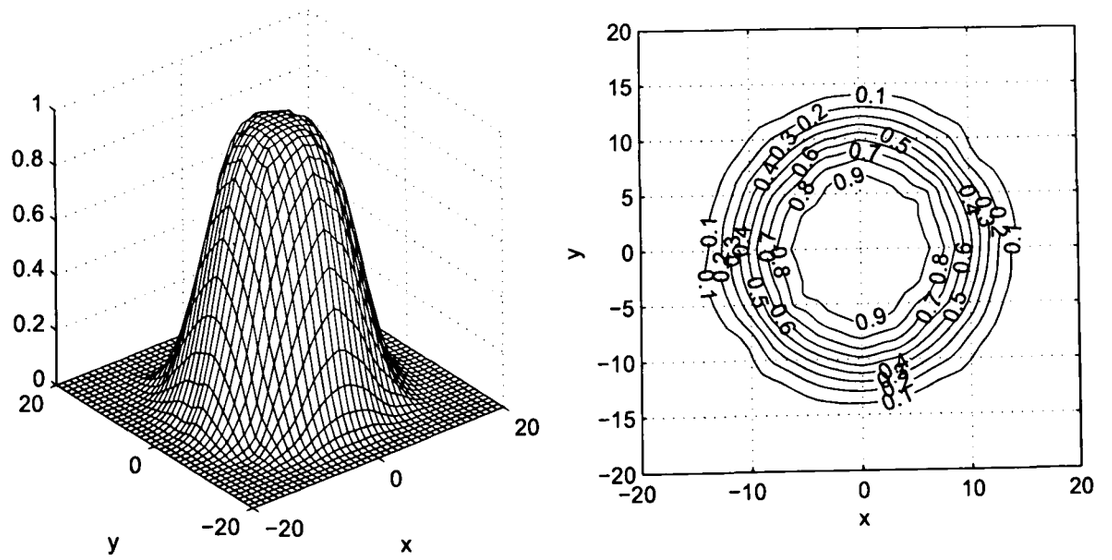


Figure 4.30: 2D PSF assuming a Generalised Gaussian model for $z = 0.725$ m and f/2.8 where x and y are in pixels

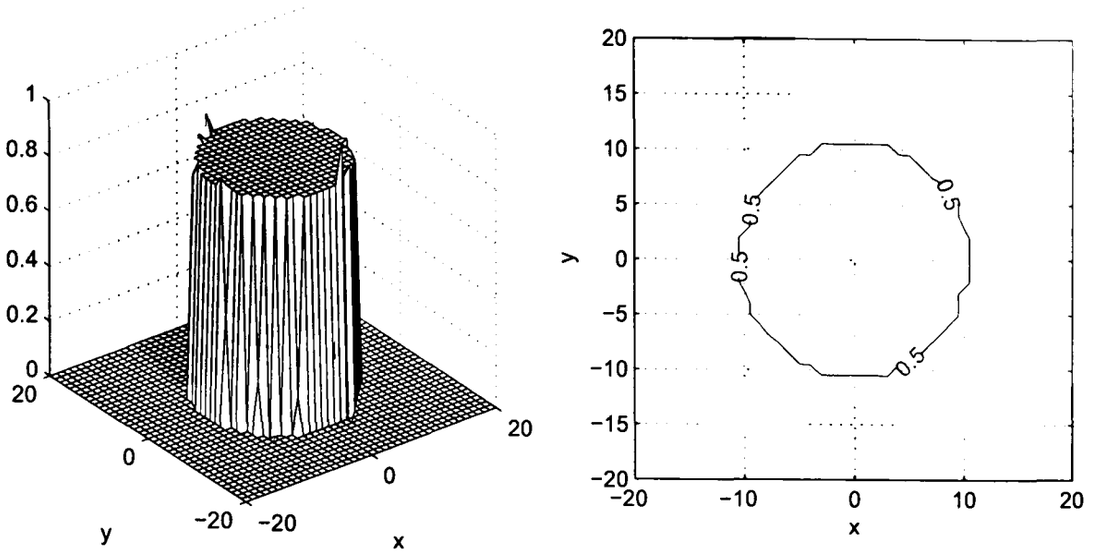


Figure 4.31: 2D PSF assuming a Pillbox model for $z = 0.725$ m and f/2.8 where x and y are in pixels

The Gaussian PSF model shown in Figure 4.29 is for the maximum distance tested, i.e. 0.725m, with an aperture of f/2.8 (the widest in the tests) and it is clearly very circularly symmetric and the fit has resulted in a smooth contour plot. The Generalised Gaussian PSF model shown in Figure 4.30 appears to be a cross between the Gaussian and a pillbox. The fit has resulted in a contour plot that is less smooth than for the Gaussian, which is probably due to noise in the ESFs and increased complexity of the function due to having more parameters than all the other models.

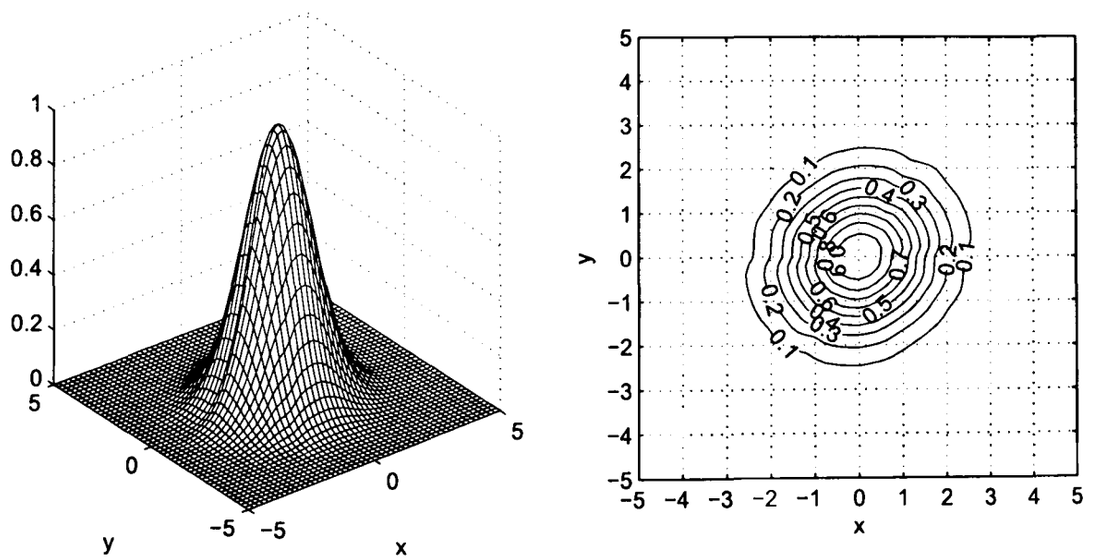


Figure 4.32: 2D PSF assuming a Gaussian model for $z = 0.414$ m and f/2.8 where x and y are in pixels

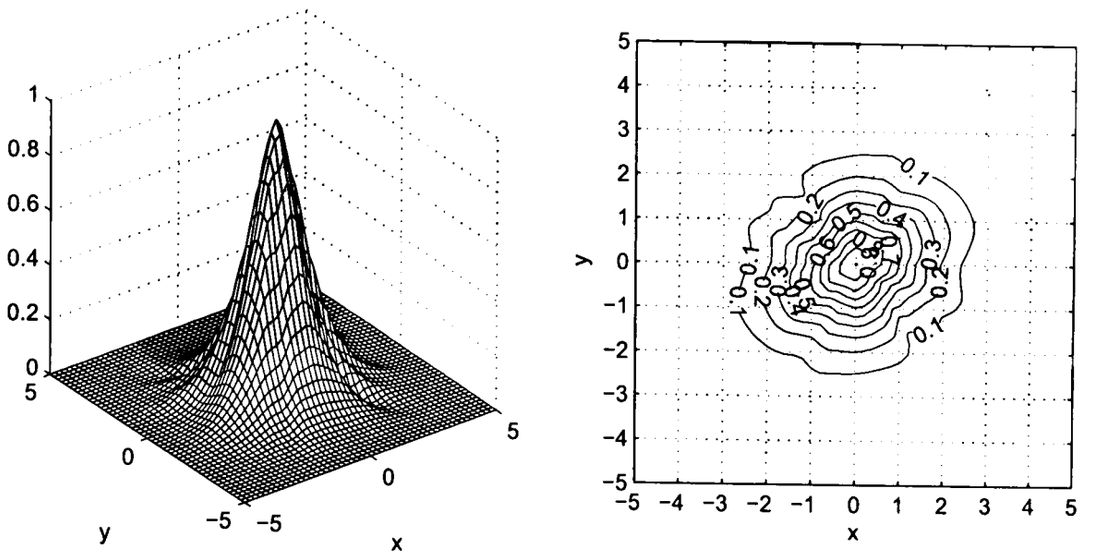


Figure 4.33: 2D PSF assuming a Generalised Gaussian model for $z = 0.414$m and f/2.8 where x and y are in pixels

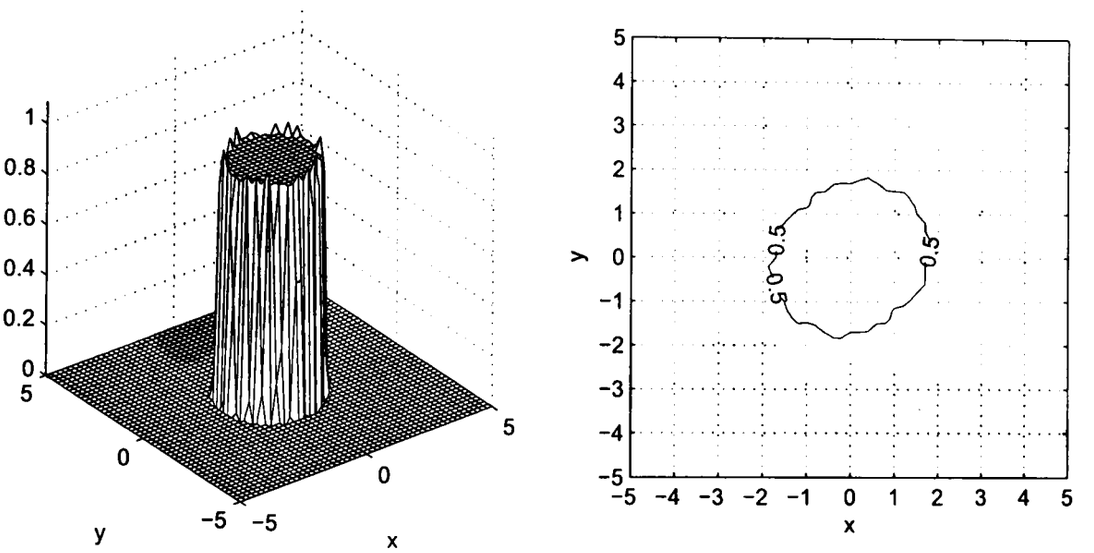


Figure 4.34: 2D PSF assuming a Pillbox model for $z = 0.414$m and f/2.8 where x and y are in pixels

Note the change of x and y axis scale in Figures 4.32 to 4.34 compared to those in Figure 4.29 to 4.31. All three models have less circularly symmetry for the closest depth of 0.414m and have a maximum spread at approximately 45 degrees to the x axis. The power of the Generalised Gaussian is less than two, and so the function is more pointed than a Gaussian. From the results of Figure 4.27 it can be seen that as the radius of the aperture is decreased (i.e a larger f-number) that the PSF becomes more pointed in shape.

4.7.6 Conclusion

The goodness-of-fit of the Generalised Gaussian PSF is exemplified by the results of Table 4.7 where the non-uniform illumination model was employed. The fit was between 9 and 16 times better than using a Gaussian PSF and on average the Generalised Gaussian model had a MSE that was 12 times better than the Gaussian model.

Table 4.7. The average MSE for each method

	Average Mean Square Error (MSE) / 10^{-3}		
Method, direction	f/2.8	f/4	f/5.6
Gaussian, x-direction	31.7	21.9	23.3
Gaussian, y-direction	46.1	27.3	23.7
Generalised Gaussian, x-direction	2.20	1.67	1.42
Generalised Gaussian, y-direction	4.99	2.44	1.87

The Gaussian PSF has a faster roll-off when the camera is very defocused compared to that using the Generalised Gaussian because the power of the Generalised Gaussian increases with defocus, thus making it more pillbox in shape, as highlighted in Figure 4.36.

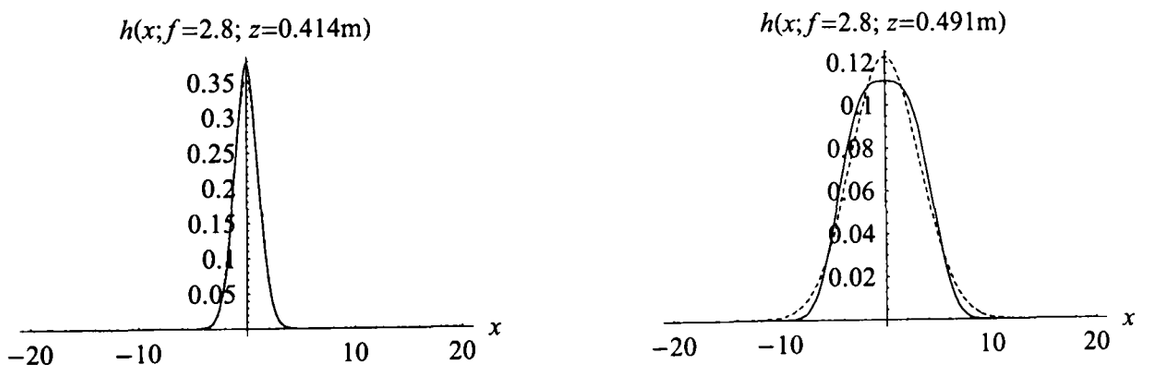


Figure 4.35: Comparison of the Gaussian (dashed line) and Generalised Gaussian (solid line)

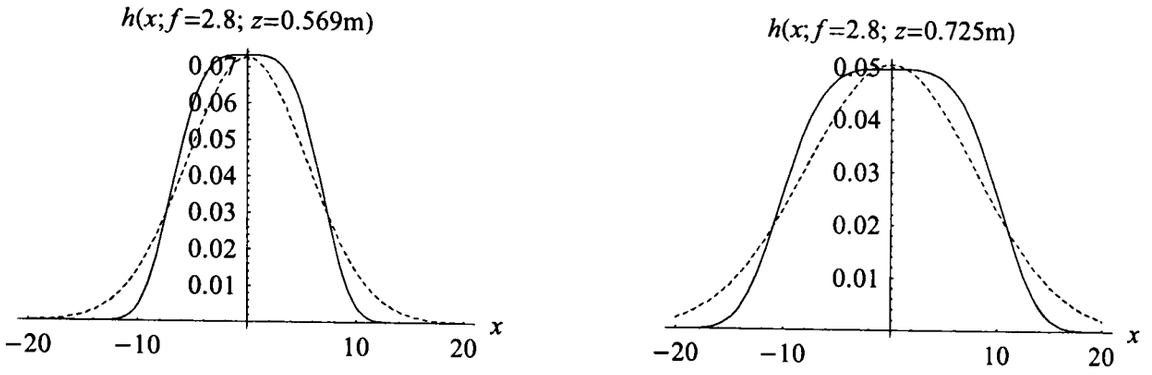


Figure 4.36: Comparison of the Gaussian (dashed line) and Generalised Gaussian (solid line)

4.8 Conclusion

The linearity of the camera is important for DFD and PSF recovery work and it was found, using the circuit designed, that the output of the camera is very linear with brightness. The bias and dark frame noise experiments showed that most of the noise was due to the readout electronics, which manifested itself as an offset brightness. The mean brightnesses can be simply subtracted from each colour plane to ensure linearity.

An automated x-stage was constructed that efficiently allows for the collection of many images for processing to find the PSF at various distances. Once the images had been demosaiced they could be used to determine an average PSF of the lens for a given lightbox angle. The form of the step was improved from Staunton's [57] original work to include non-uniform illumination. Various theoretical PSF and ESF models were proposed including Fermi-Dirac, Gaussian, Generalised Gaussian and pillbox models.

The results from the 24mm Sigma photographic lens showed that the Generalised Gaussian, Gaussian and pillbox MSEs were reduced by 27.3%, 25.0% and 26.2% respectively by incorporating the non-uniform illumination, which is clearly a significant improvement.

Pillbox and Gaussian models are often assumed in DFD work and this research has shown that both are sub-optimum. The results from the 24mm lens showed that the MSE of the fit using the Generalised Gaussian performed best across the range of distances and f-numbers tested and it was 8 times better than the Gaussian model and 14 times better than the pillbox model.

Chapter 5

The Theory of Colour Depth-From-Defocus

5.1 Introduction

Depth-from-defocus (DFD) algorithms have previously been developed for monochrome images and this chapter discusses different pre-processing algorithms that can be applied to colour images to convert them to monochrome with the aim to produce improvements in the depth maps.

Ens and Lawrence's [58] [59] algorithm was used as the basis of the research because it allows experimentally determined PSFs to be employed (which should lead to more accurate depth maps compared to resorting to a theoretical model), it is easily implemented and the results they reported were good compared to other methods developed (see the comparison in Section 2.6). As a pre-processing method, it is hoped that the results are not dependent on the particular DFD algorithm chosen and thus improvements would be obtained with other DFD algorithms too.

The errors in a generic DFD system are from:

- Noise
- Windowing and the image overlap effects
- Lack of texture
- Sub-optimum knowledge of the Point Spread Functions (PSFs)
- Software implementation

The last of the errors was reduced through the work on measuring the PSF presented in Chapters 3 and 4. The software was written in MATLAB and each function had an associated *test harness* in an attempt to reduce problems with the implementation. The hypothesis of the research presented here was that a colour imaging system can help to alleviate the remaining three problems. The multichannel DFD problem was tackled using an

implicit approach where the colour channels were compressed to a single channel using a linear combination of the colour planes, which has been called *colour mixing*.

A textureless surface does not show any change in texture with defocus and so texture is clearly an important aspect of DFD. The texture cannot be changed using a monochrome image but a black-and-white image formed from a linear combination colour planes allows for limited changes in the texture. Yuan and Subbarao [83] suggested using the band with the highest contrast, but this is not optimum. This chapter introduces the use of Principal Component Analysis (PCA) for determining the optimum scaling parameters based on a statistical analysis of the texture. The fractal dimension (FD) can be employed as a measure of the roughness of the brightness variation of a texture and a method of maximising the FD is discussed that uses spectral analysis.

Noise is an inevitable consequence of a real imaging system and a measure of the image quality is given by the signal-to-noise ratio (SNR). Maximising the SNRs of the images used in DFD was expected to result in more accurate depth maps and a method is presented that produces a monochrome image with the maximum SNR using colour mixing and an additive noise model.

Although it appears to be a paradox, in order to calculate the depth of a point in the scene a window must be applied that could have 1024 pixels in it, for a 32×32 window, or maybe even more. The finite region is required to accurately determine how the point has been blurred, but consequently the surrounding regions overlap and alter the depth estimate. A colour mixing algorithm that works on a specially designed texture is presented that aims to reduce the windowing and image overlap problem.

In Section 5.2 Ens and Lawrence's original DFD algorithm is discussed along with possible modifications to the error measurement and how multiple images could be incorporated. The concept of colour mixing as a pre-processing stage is discussed in Section 5.3 and then the different colour mixing algorithms are described in Sections 5.4 to 5.8. Finally, the chapter is summarised in Section 5.9.

5.2 Ens and Lawrence's DFD Algorithm

5.2.1 Introduction

Many different DFD algorithms have been developed, as shown by the literature review of Chapter 2. Of all the possible algorithms to build on Ens and Lawrence's look-up table based algorithm was chosen because it can readily accept experimentally determined PSFs that were found and reported in Chapter 4. It is an elegant, spatial-domain approach and the simplicity of the lookup table is attractive from an implementation point-of-view, although it is certainly not trivial.

The theoretical background to Ens and Lawrence's DFD algorithm is presented in Section 5.2.2. It was noted that they did not discuss the error measure they employed in their papers and Section 5.2.3 presents two different error measures that could be used. Section 5.2.4 presents a normalisation procedure to compensate for the exposure changes.

5.2.2 Algorithm Description

Introduction

Consider the image $f(x, y)$ that would be formed on the image plane of an ideal pinhole camera, i.e. where there are no diffraction effects, and (x, y) are the orthogonal spatial coordinates of a point on that plane. This image is often called the *focused image* in DFD because every point is in focus and although it is not physically realisable, it aids in developing algorithms. Now consider a camera with parameter set k where k is an integer and specifies the particular combination of settings, namely the focal length, aperture and focus position. If the camera is used to image the same scene $f(x, y)$ instead of a pinhole then the resultant image is given by

$$i_k(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi, \eta)\, h_k(x, y, \xi, \eta)\, d\xi\, d\eta \qquad (5.1)$$

and this represents an image defocused by the space-varying kernel $h_k(x, y, \xi, \eta)$, which corresponds to the blurring at position (x, y) as a result of the brightness at (ξ, η). The infinite limits in the integral have been left for generality, but clearly an image will have a finite spatial extent. If the depth is constant then the integral reduces to the convolution integral, given by

$$i_k(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi, \eta)\, h_k(x-\xi, y-\eta)\, d\xi\, d\eta \tag{5.2}$$

and this can be written simply as

$$i_k(x, y) = f(x, y) * h_k(x, y) \tag{5.3}$$

where $*$ denotes the operation of linear convolution and $h_k(x, y)$ is the space-invariant blurring kernel known as the Point Spread Function (PSF). Thus (5.3) represents the equation for blurred image k based on the pinhole image $f(x, y)$ and the PSF $h_k(x, y)$.

One defocused image (i.e. $k = 1$) is sufficient to determine the depth of objects as long as strong assumptions can be made about the scene, such as a known sharp intensity change in the region of interest or a projected pattern is present, as discussed in Section 2.3 in the literature review. When the content of the scene is unknown a more advanced approach is to take two images of the same scene with different camera parameters so that the contribution due to the scene can be factored out. The camera parameters that can be changed between images are the aperture size (f-number), focal length or the distance between the lens and image plane. Often just one parameter is modified, but Subbarao showed that all three could be changed simultaneously [4]. With two images $k = 1, 2$ and the equations of the two defocused images are given by

$$i_1(x, y) = f(x, y) * h_1(x, y) \tag{5.4}$$

$$i_2(x, y) = f(x, y) * h_2(x, y) \tag{5.5}$$

where the assumption of space-invariance has again been assumed and changing the camera parameters has resulted in two PSFs, $h_1(x, y)$ and $h_2(x, y)$. Further, the scene is assumed to remain unchanged between the images, there is no movement in the objects, the cameras share the same optical axis and there are no magnification changes between the images.

Ens and Lawrence's Algorithm

Ens and Lawrence [58] [59] formulated the DFD problem as that of determining the optimum *convolution ratio* $h_3(x, y)$ from the set of convolution ratios stored in the lookup table such that the least blurred image $i_1(x, y)$ convolved with $h_3(x, y)$ is the same as the most defocused image $i_2(x, y)$, i.e.

$$i_1(x, y) * h_3(x, y) = i_2(x, y). \tag{5.6}$$

The PSFs $h_1(x, y)$ and $h_2(x, y)$ are a function of the camera parameters and the depth of the object. For a given object depth, Ens and Lawrence showed that the convolution ratio is directly related to the depth and it is important that the function is monotonic and one-to-one for the depth range considered. For example, when changing the aperture only it is

important that the objects are either all in front or all behind the point of focus. Expanding (5.6) using (5.3) and (5.4) gives

$$[f(x, y) * h_1(x, y)] * h_3(x, y) = f(x, y) * h_2(x, y) \tag{5.7}$$

and so rearranging yields

$$f(x, y) * [h_1(x, y) * h_3(x, y) - h_2(x, y)] = 0. \tag{5.8}$$

The trivial solution of (5.8) is $f(x, y) = c$ where c is a constant and the scene has no intensity information and is thus useless for DFD. Of all the convolution ratios in the lookup table the particular $h_3(x, y)$ where

$$h_1(x, y) * h_3(x, y) - h_2(x, y) = 0 \tag{5.9}$$

determines the object's depth. The reason for the name *convolution ratio* can be seen by transforming (5.9) to the Fourier frequency domain where spatial domain convolution becomes frequency domain multiplication, thus

$$H_1(u, v) H_3(u, v) = H_2(u, v) \tag{5.10}$$

where $h_k(x, y) \overset{\text{FT}}{\longleftrightarrow} H_k(u, v)$ for $k = 1, 2, 3$, and so

$$H_3(u, v) = \frac{H_2(u, v)}{H_1(u, v)}. \tag{5.11}$$

Hence $h_3(x, y)$ is the inverse Fourier transform of the ratio of the Fourier transforms of the PSFs.

Due to the unavoidable presence of noise no pre-computed convolution ratio will allow the equality to exist and so an error measure must be employed. The distance measure Ens and Lawrence used was the sum of the L_2-norms (squared error) and so the problem becomes that of finding the convolution ratio $h_3(x, y)$ to minimise

$$\min \sum_{x,y} (i_1(x, y) * h_3(x, y) - i_2(x, y))^2 \tag{5.12}$$

Ens and Lawrence's algorithm tests every pre-computed convolution ratio for a given image window and measures the error in the fit. The depth resolution is determined by the choice of convolution ratios in the lookup table and if, for example, the lookup table was populated with functions with centimetre spacing then the depth map would have a minimum error of $z \pm 0.5$ cm. By adding more entries to the lookup table the potential depth resolution would increase, but the processing time would also increase.

The Causes of Under- and Over-Estimating the Depth

If the DFD system is only limited by the depth quantisation levels of the lookup table then it is working very well indeed. In practice, the DFD algorithm can produce an incorrect depth estimate and this is due to selecting the wrong convolution ratio. If the depth estimate is not correct then the only two possibilities are that the depth has been over-estimated or under-estimated.

If the depth has been over-estimated then the convolution ratio has a spatial extent that is too large. Let the erroneous convolution returned by the DFD be denoted $\tilde{h}_3(x, y)$ and the actual convolution ratio be denoted by $h_3(x, y)$. They can be linked through

$$\tilde{h}_3(x, y) = h_3(x, y) * \zeta(x, y) \tag{5.13}$$

where $\zeta(x, y)$ is a Gaussian function with a spread σ_ζ that is essentially the error in the convolution ratio. If the standard deviations of the Gaussians of $h_3(x, y)$ and $\tilde{h}_3(x, y)$ are denoted σ_3 and $\tilde{\sigma}_3$ then it can be shown that (Appendix D)

$$\tilde{\sigma}_3^2 = \sigma_3^2 + \sigma_\zeta^2 \tag{5.14}$$

and for the particular image region that produced an over-estimate of the depth

$$i_1(x, y) * \tilde{h}_3(x, y) = i_2(x, y). \tag{5.15}$$

This can be written as

$$[f(x, y) * h_1(x, y)] * \tilde{h}_3(x, y) = [f(x, y) * h_2(x, y)] \tag{5.16}$$

and substituting (5.13) into (5.16) gives

$$[f(x, y) * h_1(x, y)] * h_3(x, y) * \zeta(x, y) = [f(x, y) * h_2(x, y)]. \tag{5.17}$$

Equation (5.17) can be rearranged to give

$$[f(x, y) * h_1(x, y) * \zeta(x, y)] * h_3(x, y) = [f(x, y) * h_2(x, y)]. \tag{5.18}$$

Therefore, the depth will be over-estimated if the spread of the PSF of camera 1 was under-estimated in the camera calibration stage. This is due to the fact that the PSF $h_1(x, y)$ must be convolved with $\zeta(x, y)$ in order to give the correct depth so that

$$\left[f(x, y) * \tilde{h}_1(x, y)\right] * h_3(x, y) = [f(x, y) * h_2(x, y)] \tag{5.19}$$

where $\tilde{h}_1(x, y)$ is the correct PSF, given by $\tilde{h}_1(x, y) = h_1(x, y) * \zeta(x, y)$.

Alternatively, consider perfect camera calibration, so that (5.17) can be simplified to

$$i_1(x, y) * h_3(x, y) * \zeta(x, y) = i_2(x, y) \tag{5.20}$$

using $i_1(x, y) = f(x, y) * h_1(x, y)$ and $i_2(x, y) = f(x, y) * h_2(x, y)$. Rearranging (5.20) gives

$$[i_1(x, y) * \zeta(x, y)] * h_3(x, y) = i_2(x, y). \tag{5.21}$$

Thus, another cause of over-estimating the depth occurs when the image $i_1(x, y)$ must be smoothed to make $i_1(x, y) * h_3(x, y) = i_2(x, y)$. This would be the case if image 1 is too noisy, i.e. the high frequency content must be reduced to ensure a perfect depth estimate. Alternatively, image 2 must be too smooth, i.e. its high frequency content is too low.

Now consider the other case where the depth has been under-estimated, thus the spatial extent of the optimum convolution ratio is too small. The modification (5.13) cannot be used because convolution of the actual $h_3(x, y)$ with $\zeta(x, y)$ cannot reduce the spread, as can be seen from (5.14). Therefore, (5.16) must be changed to

$$[f(x, y) * h_1(x, y)] * h_3(x, y) = [f(x, y) * h_2(x, y)] * \zeta(x, y) \tag{5.22}$$

and this can be rearranged to give

$$[f(x, y) * h_1(x, y)] * h_3(x, y) = f(x, y) * [h_2(x, y) * \zeta(x, y)] \tag{5.23}$$

and so it can be seen that the depth is under-estimated if the spread of the PSF of camera 2 is under-estimated and it must be corrected by convolution with $\zeta(x, y)$. If there is no error in the camera calibration then (5.22) reduces to

$$i_1(x, y) * h_3(x, y) = i_2(x, y) * \zeta(x, y) \tag{5.24}$$

and thus in order to produce the correct depth, $i_2(x, y)$ must be smoothed, thus reducing its high frequency content. Hence, too much noise in $i_2(x, y)$ will cause the depth to be under-estimated. Alternatively, image 1 is too smooth. A further cause of the depth being under-estimated is if there are two objects in the window at different distances and the closer object is giving an undesired contribution to the intensity, i.e. the depth of the further object is required.

In summary, the depth is over-estimated if:

- The spread of the PSF of camera 1 is under-estimated;
- Image 1 has too much high frequency content (due to noise for example);
- Image 2 is too smooth (i.e. too little high frequency content)

The depth is under-estimated if

- The spread of the PSF of camera 1 is under-estimated;
- Image 2 has too much high frequency content (due to noise or an object in the window contributing too much high frequency information, for example);
- Image 1 is too smooth.

The depth error is also dependent on the error measure employed and this is the subject of the next section.

5.2.3 The Error Measurement

The less defocused image $i_1(x, y)$ blurred with the convolution ratio $h_3(x, y)$, denoted by $\hat{i}_2(x, y) = i_1(x, y) * h_3(x, y)$, is an approximation of the more blurred image $i_2(x, y)$. Ens and Lawrence chose to use the sum of the L_2-norms so that the convolution ratio that results in the minimum sum of squared differences determines the depth. Other measures that could have been employed include the sum of the L_1-norms (total variation), given by

$$\varepsilon = \sum_{x,y} \left| \hat{i}_2(x, y) - i_2(x, y) \right| \tag{5.25}$$

and the information-divergence (I-divergence) proposed by Csiszár [97] and based on work by Kullbach [133], which is given by

$$I\left(\hat{i}_2 \,\|\, i_2\right) = \sum_{x,y} \left(\hat{i}_2(x, y) \log\left(\frac{\hat{i}_2(x, y)}{i_2(x, y)} \right) - \hat{i}_2(x, y) + i_2(x, y) \right). \tag{5.26}$$

The I-divergence is a error measure between two non-negative functions from an information-theoretic point of view. It was used by Favaro and Soatto [96] in their work on DFD. As a theoretical analysis could not be undertaken, an empirical approach was employed and the results are presented in the next chapter. The sum of the L_2-norms emphasises the large errors, where the difference is greater than 1, and de-emphasises errors in the range [0, 1]. In contrast the sum of the L_1-norms does not emphasise the large variations, but penalises small errors [134].

5.2.4 Normalisation of the Image Segments

Ens and Lawrence's formulation in (5.12) is based on the irradiance of the scene being identical between images and the change in f-number being compensated for. By using two widely different f-numbers the exposure time must be changed to ensure that the less defocused image taken with the smallest aperture is not buried in noise while the image taken with the widest aperture is not saturated. To simplify the work the most defocused image segment $i_2(x, y)$ and its approximation $\hat{i}_2(x, y) = i_1(x, y) * h_3(x, y)$ were normalised to be in the range [0, 1] using

$$i_{2_N}(x, y) = \frac{i_2(x, y) - \min[i_2(x, y)]}{\max[i_2(x, y)] - \min[i_2(x, y)]} \tag{5.27}$$

$$\hat{i}_{2_N}(x, y) = \frac{\hat{i}_2(x, y) - \min\left[\hat{i}_2(x, y)\right]}{\max\left[\hat{i}_2(x, y)\right] - \min\left[\hat{i}_2(x, y)\right]}. \tag{5.28}$$

where min[X] and max[X] are the minimum and maximum intensities of image X. Therefore, the intensities of i_{2_N} and $\hat{i}_{2_N}$ all lie in the range [0,1].

5.2.5 Conclusion

The accuracy of Ens and Lawrence's DFD algorithm based on a look-up table is dependent on how well the PSFs were modelled, as with so many of the DFD algorithms reviewed in Chapter 2. The algorithm was based on the assumption that the depth is constant within a window and that there is sufficient texture from which to measure the change in defocus between the two images used. This section has presented a logical development of Ens and Lawrence's DFD algorithm and has shown that the particular error measure must be evaluated in practice. A simple normalisation of the image segments has been presented that accounts for the differences in brightnesses due to the two apertures.

5.3 Colour Mixing as a Pre-Processing Stage

5.3.1 Introduction

The emission spectra of the illumination source and the effects of wavelength-dependent absorption, refraction, diffraction and scattering of objects in the scene coupled with the response of the human visual system leads to the appearance of colour [135]. A monochrome camera produces an output signal that is dependent on the number of photons arriving at a given photosite. No colour filter is employed, such as a CFA, and thus the response is a function of the spectral content of the light, the quantum efficiency of the detector for a given wavelength (and thus colour) and the attenuation due to the lens elements and any coatings applied. An RGB colour camera captures three intensity images in three different bands of the visible spectrum that are generally overlapping.

A monochrome image $M_k(x, y)$ can be formed from the three colour planes through a linear operation and it is expressed as

$$M_k(x, y) = \alpha_k R_k(x, y) + \beta_k G_k(x, y) + \gamma_k B_k(x, y) \tag{5.29}$$

where $R_k(x, y)$, $G_k(x, y)$ and $B_k(x, y)$ are the red, green and blue planes respectively of image k, (x, y) are the orthogonal spatial coordinates and $(\alpha_k, \beta_k, \gamma_k)$ are real scaling constants. A standard measure of intensity of an image is given setting $\alpha_k = \beta_k = \gamma_k = \frac{1}{3}$ in (5.29).

Depth-from-defocus requires two images so that the content due to the scene can be factored out and generally there will be two sets of scaling coefficients, $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$. It is shown in Appendix B that it is important that the colour planes of both images are scaled identically, i.e. $\alpha_1 = \alpha_2$, $\beta_1 = \beta_2$, and $\gamma_1 = \gamma_2$.

The dimension reduction using (5.29) was the basis of the research with the question being whether more accurate depth maps could be produced by choosing (α, β, γ) to meet some criteria instead of using $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$. This approach to colour image processing is not new and it has been used in forensic and the processing of satellite imagery.

Berger *et al.* [35] devised a colour mixing algorithm to enhance the required colours of a document to ascertain whether additions had been made to handwriting and also unmasking text that had been covered with ink for example.

Multichannel satellite images are processed to reveal information about the Earth's surface. In particular, NASA's Landsat images are used for discriminating crop types, mapping geological structures and monitoring coral reefs and volcanic activity. A common problem is to produce a single channel image with the most information from a linear combination of six or maybe more satellite image channels such as RGB and near-, mid- and far-infrared.

The layout of this Section is as flows. In Section 5.3.2 the use of colour filters in black-and-white photography is first explored before examining how colour mixing is an approximation to applying physical colour filters in the optical path of the camera in Section 5.3.3. Hue, saturation and intensity are important for describing colours and Section 5.3.4 examines when colour mixing can be done by considering the HSI space.

5.3.2 The Use of Colour Filters in Black-and-White Photography

Colour and polarising filters have been employed by black-and-white photographers for decades and nowadays, artistic effects can be applied in digital photography software. Polarising filters are used universally by digital and film photographers as they have the effect of dramatising sky and clouds in a scene because light coming from a clear sky is polarised, with the greatest effect at ninety degrees to the sun [136].

Ultra-violet (UVa) filters were originally employed to cut out UVa in the atmosphere so that clearer photographs could be captured [136], but now they are usually employed to protect lenses as they are relatively inexpensive and many lenses now have built-in UV filters.

Neutral density (ND) filters are designed for the purpose of equally attenuating the light entering the camera over all wavelengths of visible light and thus they possess a grey

colour. The colourless tone ensures that they do not affect the colour balance [136] and the densities are specified by a factor, e.g. 2, 4, etc. ND filters are particularly useful for blurring motion (such as flowing rivers) in daylight where the filter allows an increase in the shutter time. They allow a wider aperture for a given scene (such a flower) which reduces the depth-of-field, the effect being to just keep the subject in focus with a blurred background.

Colour filters are equally useful in black-and-white and colour photography, their purpose being to attenuate desired frequencies of light; and note that they do not add colour to an image. A colour filter passes their own colour well and attenuates (darkens) the complementary colour, which can be found from a colour wheel.

There are many different colour filters employed in photography. For example, a blue filter is used in medical imaging to produce good contrast between blood vessels and scars [136]. Filters used in photography either have a constant colour or possess a gradient so that the colour effect changes smoothly. Physical colour filters can be used in the optical path or a similar effect can be achieved in software, but the effect is not the same, as shown in the theoretical analysis below.

5.3.3 Why Physical Filters are Superior to Digital Colour Mixing

Consider a device, for example a CCD or cones on the human retina, that have an absorption spectra $S_i(\lambda)$ where the integer i denotes the specific colour response and λ is the wavelength of light. For the human retina i will be in the range [1, 3] as there are three different types of cones. The response $R_i(C)$ of sensor i to the light with a spectral distribution of $C(\lambda)$ is given by [34]

$$R_i(C) = \int_0^\infty S_i(\lambda)\, C(\lambda)\, d\lambda \tag{5.30}$$

where an infinite limit has been used for generality, but $S_i(\lambda)$ and $C(\lambda)$ will be bandlimited. Suppose now a semi-transparent colour filter with a spectral transmittance distribution $F(\lambda)$ is placed in front of the sensor. The response of the sensor will now be

$$R_i(C) = \int_0^\infty F(\lambda)\, S_i(\lambda)\, C(\lambda)\, d\lambda. \tag{5.31}$$

This research considered the colour mixing of images that were taken by a camera that did not have a filter applied, as firstly it is unlikely that the optimum colour is known beforehand and secondly, it is likely to require a colour that is a function of spatial coordinates. As

$$\int_0^\infty F(\lambda)\, S_i(\lambda)\, C(\lambda)\, d\lambda \neq \int_0^\infty F(\lambda)\, d\lambda \int_0^\infty S_i(\lambda)\, C(\lambda)\, d\lambda \tag{5.32}$$

then the integrated response of the filter $F(\lambda)$ multiplied by response of sensor i to $C(\lambda)$ is not the same as the response of the sensor with the filter in front (the left hand side of the equation). Thus, the exact spectral response of the sensor cannot be reproduced by scaling the colour planes.

It is impossible to recover the spectrum of the light $C(\lambda)$ from the samples $R_i(C)$ by virtue of the loss of information due to the integration. Consider that any two colours $C_1(\lambda)$ and $C_2(\lambda)$ where $C_1(\lambda) \neq C_2(\lambda)$ such that $R_i(C_1) = R_i(C_2)$ for all i. The colours will be perceived to be identical and the colours are termed *metamers*, even though they are spectrally dissimilar [34].

A three-colour camera either employs two beam splitters, three colour filters and three CCDs or uses a single CCD and a Colour Filter Array (CFA), such as a Bayer filter where each pixel is covered by either a red, green or blue filter. Both camera systems suffer from a severe reduction in the spectral information as knowledge of $C(\lambda)$ cannot be regained. If many colour filters could be used each with a narrow pass-band then less information would be lost.

Even though only an approximation to physical colour filters can be achieved through adjusting the quantities of the red, green and blue components returned using a 3-colour camera there is limited scope for change. A more complete spectral representation would facilitate a greater adjustment.

5.3.4 Colour Spaces and Colour Mixing

A few different colour spaces were discussed in Chapter 1. One particularly useful colour space for describing colours is Hue-Saturation-Intensity (HSI). In order to understand the link between the HSI space and colour mixing an analysis was performed. Appendix C gives a derivation of the fact that if an image has a change in intensity but constant hue and saturation then colour mixing using (α, β, γ) is no different from using $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$. In contrast an image with hue and saturation variations allows for limited colour mixing.

5.3.5 Colour Mixing and Depth-From-Defocus

The literature survey revealed that many monochrome DFD algorithms had been developed, but there was not one algorithm that specifically used colour images. It is known that colour image of a scene possess more information than a corresponding monochrome image owing the increased number of bands and so the aim of the research was to investigate if there were benefits in terms of increased depth accuracy using colour images.

The specific problems of a generic DFD system listed in the Introduction were investigated through the framework of colour mixing. In the initial stages of the research, a genetic algorithm was employed to evolve the optimum scaling coefficients (α, β, γ) to reduce the depth error of a known scene with a known depth map. This showed that there was merit in using colour mixing and so deterministic algorithms were then sought to solve the problems listed.

In Section 5.4 to 5.8, the theory behind the different approaches to colour mixing that were investigated are presented.

5.3.6 Conclusion

This section has introduced the concept of employing a linear combination of colour planes to produce a monochrome image with the problem being to determine the optimum combination, and this is left to the next sections. Physical colour filters are superior to using colour mixing, however, this approach is not practical unless the environment is very carefully controlled. The HSI analysis has showed that changes in the hue and saturation of a colour texture are required.

5.4 Initial Genetic Algorithm Research

5.4.1 Colour Mixing with a Known Depth Map

The basis for the research into a colour depth-from-defocus algorithm began with the realisation that a scene could theoretically appear textureless to a monochrome camera, but in fact could be composed of many colours, and thus possess texture in the spectral dimension. Consider for example the very simple scene composed of a 4×4 grid of pixels where the red, green and blue components are given by

$$\mathbf{R} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix} \tag{5.33}$$

and using equal contributions due to all three colour planes results in the monochrome image

$$\mathbf{M} = \frac{1}{3}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \tag{5.34}$$

using (5.29) with $\alpha = \beta = \gamma = \frac{1}{3}$. Image $\mathbf{M}$ is clearly textureless as each pixel has the same intensity and is thus useless for DFD. In contrast each individual colour plane shows intensity variations and would perform better. If the surface was grey in colour then each colour plane will have a very limited variation in brightness and so it is less useful than a colour surface for DFD.

An optimisation algorithm was used to find the best (α, β, γ) to minimise the depth error using DFD and the results are presented in Section 6.3 of the next chapter. The optimisation was performed using a Genetic Algorithm that evolves the solution to the problem in analogy to biological evolution using the principle of *Survival of the Fittest* [137]. A population of individuals are randomly generated at the start and each individual is represented in the computer as a long binary number, which ultimately maps to a particular (α, β, γ). Each individual is tested by scaling the colour planes and then running the resulting monochrome image through the DFD algorithm. The depth error is calculated and the individual is then assigned a fitness value based on how close the depth is to the actual, where a smaller depth error results in a higher fitness value. Of the population a given proportion of them are allowed to 'mate' and their probability of mating increases as their fitness value increases. As with the biological counterpart, offspring are created that have genes from both parents (formed using cross-over and mutation) and they represent new (α, β, γ) values. A certain proportion of the parent generation die off and the process continues for a set number of generations. When the final generation is reached the individual with the highest fitness is used to give the optimum (α, β, γ).

5.4.2 Colour Mixing with an Unknown Depth Map

The approach discussed in the previous section is applicable only for scenes with a known depth map and therefore it was only useful as an initial research tool. The results presented in the next chapter show that colour mixing has the potential to perform better than using a simple equal weighting of the colour planes. Deterministic approaches that optimise a given property of the image were then explored based on the problems of a generic DFD system listed in the Introduction.

Large non-uniform intensity regions are useless for DFD and thus there must exist brightness variations. In Section 5.5, PCA is discussed and it is a standard technique for producing decorrelated colour planes, one of which possesses the maximum variance.

Whereas PCA is based on the variance of the texture, the fractal dimension is a measure of its roughness. The concept of FD is discussed in Section 5.7 along with an evolutionary method to increase the roughness of a monochrome image through colour mixing.

Noise in a DFD system will clearly adversely affect the depth map accuracy and so a method of increasing the SNR was sought. In Section 5.6 a method for maximising the SNR using colour mixing is presented.

The problem with windowing effects was discussed in Section 2.2 in relation to the matching or correspondence problem. In Section 5.8 a theoretical analysis of an active DFD method to improve localisation and thus decrease the windowing problem is discussed.

5.5 Principal Component Analysis

5.5.1 Introduction

Colour images have two spatial dimensions (height and width) and a spectral dimension, which is an aggregate response of the wavelength-dependent photodetectors. One approach to performing DFD on colour images is to compress the images down to possess just one spectral dimension and this can be achieved using mixtures of the red, green and blue colour planes, for example equal weightings. This section examines an efficient technique that performs a linear transformation on the colour planes to yield a lower dimension image with maximal variance using Principal Component Analysis (PCA).

Principal Component Analysis was developed independently by Pearson [138] and Hotelling [139] and it goes by several names including the Karhunen-Loève transform and the Hotelling transform. Whereas the Fourier transform and discrete cosine transform decompose a signal into fixed bases, PCA has basis vectors that depend on the data set employed.

Consider the image shown on the left in Figure 5.1. Each pixel has an associated red, green and blue component and these can be plotted in the RGB space as shown on the right hand side of Figure 5.1. PCA transforms the RGB axes to give new orthogonal axes (bases) such that the data is uncorrelated between bases. In the figure the red, green and blue lines show the first, second and third principal axes respectively.

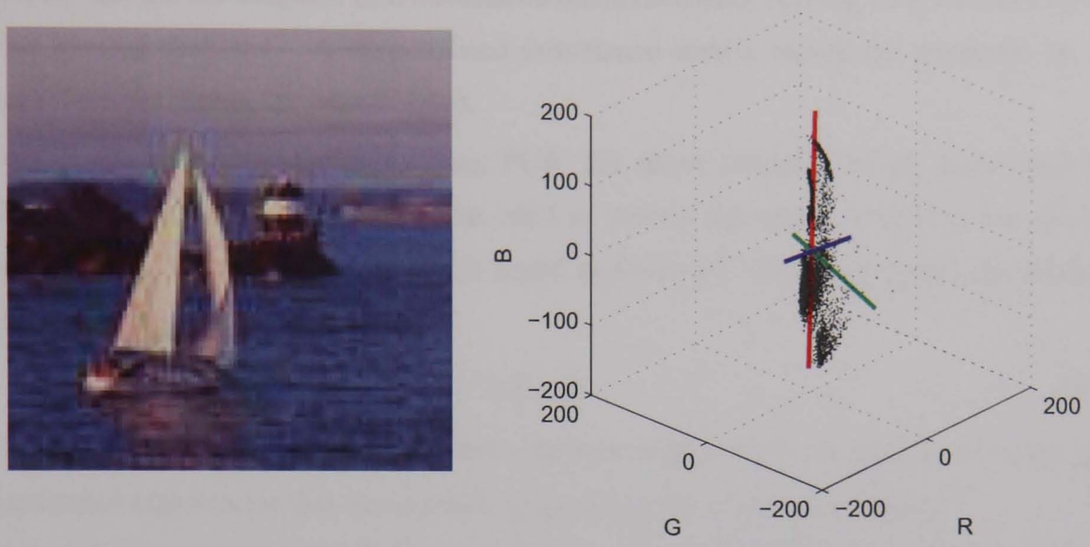


Figure 5.1: The image of a yacht (left) and the cloud of RGB points with the principal axes (right)

5.5.2 Mathematical Outline of PCA

Consider the i^{th} colour plane of an $M \times N$ image denoted x_i. The image is row-stacked to produce an $MN \times 1$ vector. A single colour plane i with MN pixels has a mean brightness associated with it, given by

$$\overline{x}_i = \frac{1}{N^2} \sum_{j=1}^{MN} x_i(j) \tag{5.35}$$

where $x_i(j)$ is the j^{th} pixel of plane i. A measure of the spread of the pixel intensities is given by the variance σ_i^2, which can be calculated using

$$\sigma_i^2 = \frac{1}{N^2} \sum_{j=1}^{MN} (x_i(j) - \overline{x}_i)^2. \tag{5.36}$$

A measure of the similarity between two colour planes is given by the covariance,

$$\sigma_{i,j} = \frac{1}{N^2} \sum_{k=1}^{MN} (x_i(k) - \overline{x}_i)(x_j(k) - \overline{x}_j) \tag{5.37}$$

and note that the covariance where $i = j$ is the simply the variance. The covariances can be placed into a matrix and for an image composed of RGB colour planes the matrix takes the form

$$\mathbf{C} = \begin{pmatrix} \sigma_R^2 & \sigma_{R,G} & \sigma_{R,B} \\ \sigma_{R,G} & \sigma_G^2 & \sigma_{G,B} \\ \sigma_{R,B} & \sigma_{G,B} & \sigma_B^2 \end{pmatrix}. \tag{5.38}$$

The order of the terms in the covariance equation does not matter and so the matrix $\mathbf{C}$ is symmetric. The goal of Principal Component Analysis is to diagonalise the covariance

matrix so that the off-diagonal (i.e. covariance terms) are zero, leaving only variance terms on the leading diagonal. A diagonalised covariance matrix would be produced by an image where the planes are uncorrelated.

This is the procedure for performing PCA: the mean intensity of the band must be subtracted from the pixel values for that band to leave a zero-mean intensity; the covariance matrix $\mathbf{C}$ of the zero-mean bands is found and then the eigenvectors and eigenvalues λ_i of the matrix are calculated from

$$|\mathbf{C} - \lambda_i \mathbf{I}| = 0 \tag{5.39}$$

where $\mathbf{I}$ is the 3×3 identity matrix. There are three eigenvalues, the largest of which has an associated eigenvector that corresponds to the direction of maximal spread.

Geometrically, if the RGB components of an image are plotted in three-dimensional space then a cloud of points will take the shape of a hyperellipsoid and the eigenvectors give the principal axes of the hyperellipsoid [140]. The eigenvectors are placed into a matrix $\mathbf{A}$ as rows and then the RGB components of a single pixel are transformed using $p = \mathbf{A}\,x$, which can be expanded to give

$$\begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{5.40}$$

where R, G and B are the intensities of a given pixel in each zero-mean colour plane. The resulting components p_1, p_2 and p_3 are orthogonal and formed from linear combinations of the colour planes. By using only the first principal component p_1 – that corresponding to the largest eigenvalue – a monochrome image is produced that has the most information [141]. The resulting space formed by the vector p is often termed the *feature space* because PCA finds statistical patterns in the data. The original data can be obtained from the transformed data using $x = \mathbf{A}^{-1} p$ and then adding on the means that were subtracted initially.

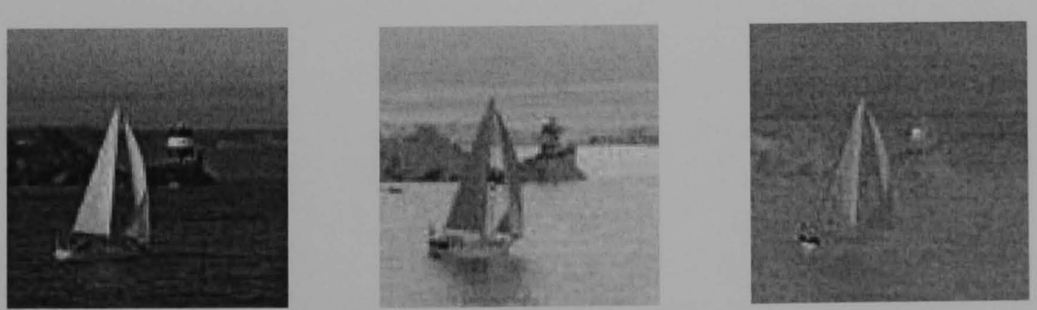


Figure 5.2: The first (left), second (middle) and third (right) principal planes of the yacht image

Thus, Principal Component Analysis finds a new orthogonal basis in which to represent the original data such that the transformed planes are uncorrelated. The choice of the

matrix that is diagonalised makes a difference to the resulting orthogonal bases and the three main possibilities are [142]:

- Covariance matrix (as used above)
- Correlation matrix
- Weighted

Weighting the colour planes could be used if there was a reason to give them ranked priority.

5.5.3 Monochrome from the Perspective of PCA

The monochrome or equal weighting algorithm uses $\alpha = \beta = \gamma = \frac{1}{3}$ and it can be analysed from the perspective of the PCA approach. Suppose the eigenvector corresponding to the largest eigenvalue, i.e. p_1, is given by (1, 1, 1) then PCA has produced the monochrome case. This means that the principal axis of the hyperellipsoid is in the direction (1, 1, 1). In the case where the spreads in the other two orthogonal components are zero, i.e. the data points lie on the line, the colour planes are maximally correlated. The image that produced this result need not be grey because the means of the colour planes are subtracted.

5.5.4 Conclusion

Principal Component Analysis is a well-established method to produce decorrelated colour planes. The principal plane with the largest variance is likely to be the optimum plane to use for DFD since the presence of texture is important. The range of the scaling parameters for PCA is $-1 \leq \alpha, \beta, \gamma \leq 1$.

5.6 Signal-to-Noise Ratio Maximisation

5.6.1 Introduction

Noise is an inevitable problem in a real camera system and it will clearly degrade the accuracy of a depth map created using DFD. Horii [5] criticised Ens and Lawrence's matrix-based approach because he believed that the technique is very dependent on the signal-to-noise ratio (SNR). One of the ways to reduce noise is to smooth the images, using a Gaussian kernel for example, but the extra smoothing increases the effective defocus, reduces the depth localisation and further reduces the crucial brightness variations that allow defocus measurements.

In this section, an additive model of the complete DFD system noise is proposed and then a solution is found to the problem of maximising the SNR through colour mixing based on finding the variance of the texture and the noise. The formulation is in keeping with the other algorithms developed and thus allows a more direct comparison. Further, the widely understood measure of the SNR using the ratio of the signal to noise variance is simple to compute.

5.6.2 Theory

The signal-to-noise ratio (SNR) is defined as [134]

$$\text{SNR} = 10 \log_{10}\left(\frac{\text{Var[signal]}}{\text{Var[noise]}}\right)\text{dB} \tag{5.41}$$

where Var[X] denotes the variance of a signal X. The noise-free monochrome image $M(x, y)$ is formed from scaled versions of the RGB colour planes so that

$$M(x, y) = \alpha R(x, y) + \beta G(x, y) + \gamma B(x, y) \tag{5.42}$$

and assuming additive noise then the colour mixed image is given by

$$M(x, y) = \alpha [R(x, y) + N_R(x, y)] + \beta[G(x, y) + N_G(x, y)] + \gamma[B(x, y) + N_B(x, y)] \tag{5.43}$$

where $N_R(x, y)$, $N_G(x, y)$ and $N_B(x, y)$ are the noise components for the red, green and blue planes respectively. The signal and noise terms can be split up to give

$$M(x, y) = [\alpha R(x, y) + \beta G(x, y) + \gamma B(x, y)] + [\alpha N_R(x, y) + \beta N_G(x, y) + \gamma N_B(x, y)] \tag{5.44}$$

and so the signal-to-noise ratio is given by

$$\text{SNR} = 10 \log_{10}\left(\frac{\text{Var}[\alpha R(x, y) + \beta G(x, y) + \gamma B(x, y)]}{\text{Var}[\alpha N_R(x, y) + \beta N_G(x, y) + \gamma N_B(x, y)]} \right) \text{dB}. \tag{5.45}$$

If $X_1, \ldots, X_N$ are random variables such that $\text{Var}[X_i] < \infty$ for all $i = 1, \ldots, N$ and a_i are constants then

$$\text{Var}\left[\sum_{i=1}^{N} a_i X_i \right] = \sum_{i=1}^{N} a_i^2 \text{Var}[X_i] + 2 \sum \sum_{i<j} a_i a_j \text{Cov}[X_i, X_j] \tag{5.46}$$

where $\text{Cov}[X_i, X_j]$ is the covariance of X_i and X_j [143].

Returning to the specific case of the colour mixing, the variance terms can be expanded to give

$$\begin{aligned} &\text{Var}[\alpha R(x, y) + \beta G(x, y) + \gamma B(x, y)] = \alpha^2 \text{Var}[R] + \beta^2 \text{Var}[G] + \\ &\gamma^2 \text{Var}[B] + 2 (\alpha \beta \text{Cov}[R, G] + \alpha \gamma \text{Cov}[R, B] + \beta \gamma \text{Cov}[G, B]). \end{aligned} \tag{5.47}$$

Note that if a single monochrome image I is present with noise I_N that scaling the image, using a constant α, cannot produce an improvement in the SNR as the scaling constants cancel,

$$\text{SNR} = 10 \log_{10}\left(\frac{\text{Var}[\alpha I]}{\text{Var}[\alpha I_N]} \right) = 10 \log_{10}\left(\frac{\alpha^2 \text{Var}[I]}{\alpha^2 \text{Var}[I_N]} \right) = 10 \log_{10}\left(\frac{\text{Var}[I]}{\text{Var}[I_N]} \right). \tag{5.48}$$

However, for an image composed of two or more colour planes it is possible to change the SNR by altering the proportions of each plane.

5.6.3 Maximisation of the SNR

The SNR assuming an additive noise model is given by (5.45) and closed-form solutions to the problem were sought, however, to no avail. One solution to the problem is to use a Genetic Algorithm to evolve the scaling coefficients (α, β, γ) to maximise the SNR, i.e.

$$\max_{(\alpha,\beta,\gamma)} 10 \log_{10}\left(\frac{\text{Var}[\alpha R(x, y) + \beta G(x, y) + \gamma B(x, y)]}{\text{Var}[\alpha N_R(x, y) + \beta N_G(x, y) + \gamma N_B(x, y)]} \right) \tag{5.49}$$

subject to $-1 \leq \alpha, \beta, \gamma \leq 1$.

The optimum (α, β, γ) are scene-dependent, as can be seen by considering the case where the camera is imaging a surface that only produces a response in one colour plane, for example the red plane. In that case, the other two planes will only consist of noise, and in the example these will be the green and blue planes. The optimum $(\alpha, \beta, \gamma) = (1, 0, 0)$ in the example because using either the green or blue plane adds noise to the resulting monochrome image. Clearly, changing the surface colour such that it appears in a different colour plane will require a new set of (α, β, γ)

5.6.4 Conclusion

The SNRs of the images are clearly an important factor for the accuracy of depth maps generated by a DFD algorithm. Averaging images taken by the camera will increase the SNR, but at the cost that the scene must remain constant, which is not a problem for static, experimental scenes. By using knowledge of the additive noise the SNR can be boosted through colour mixing.

5.7 Fractal Dimension Maximisation

5.7.1 Introduction

In this section, the presence of texture will be shown to be vitally important for DFD. In Section 5.7.2, different methods of texture analysis are reviewed and then in Section 5.7.3, the concept of fractal dimension is explored. The problem of measuring the fractal dimension is discussed in Section 5.7.4 before the colour mixing algorithm based on the fractal dimension is described in Section 5.7.5.

Consider a surface $f(x, y)$ perpendicular to the optical axis of a camera with a PSF $h_k(x, y)$. The defocused image $i_k(x, y)$ is given by

$$i_k(x, y) = f(x, y) * h_k(x, y) \tag{5.50}$$

and this can be written as

$$i_k(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi, \eta)\, h_k(x-\xi, y-\eta)\, d\xi\, d\eta. \tag{5.51}$$

where infinite limits have been employed so that boundary effects can be ignored.

Suppose the surface has a uniform radiance, i.e. no brightness variation, then $f(x, y) = a$ where a is a real constant. The defocused image becomes

$$i_k(x, y) = a\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_k(x-\xi, y-\eta)\, d\xi\, d\eta \tag{5.52}$$

and for a non-light absorbing lens, the PSF has unit volume, thus

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_k(x-\xi, y-\eta)\, d\xi\, d\eta = 1. \tag{5.53}$$

Substituting (5.53) into (5.52) gives

$$i_k(x, y) = a. \tag{5.54}$$

As discussed in Section 5.2, Ens and Lawrence's algorithm searches for the convolution ratio $h_3(x, y)$ such that

$$i_1(x, y) * h_3(x, y) = i_2(x, y). \tag{5.55}$$

For the uniform irradiance scene (5.55) becomes

$$a * h_3(x, y) = a \tag{5.56}$$

as $i_k(x, y) = a$ for $k = 1, 2$ from (5.54). Equation (5.56) can be written as

$$a \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h_3(x - \xi, y - \eta)\, d\xi\, d\eta = a \tag{5.57}$$

and since all convolution ratios $h_3(x, y)$ have a unit volume, then all convolution ratios satisfy (5.57), thus showing that the depth cannot be recovered when the surface being imaged has a uniform radiance.

The presence of brightness variations is vitally important for the DFD algorithm to operate, and this is in fact true of all DFD algorithms. The brightness variation is called visual texture [144] and the next section examines different ways of analysing textures.

5.7.2 Texture Analysis

Introduction

Texture is easily recognised by humans, but it is very difficult to define, as illustrated by the fact that there are many different definitions within literature [144]. Properties that have been used to describe textures include uniformity, density, coarseness, roughness, regularity, linearity, directionality, direction, frequency and phase [145]. The intensity variations in a scene are frequently due to the underlying physical process [144]. The rules and features that characterise a texture and local intensity variations of the associated pixels are known as texture features [146].

For any textured surface there is a scale at which the surface appears smooth and textureless and as the resolution increases it appears to have a fine texture and with a further increase of resolution it appears coarse [147]. Thus, the appearance of the texture depends on the scale of reference.

The techniques of analysing textures can be divided into statistical, geometrical, model-based and signal processing methods [144] and a very brief review of each is presented below.

Statistical Methods

A simple statistical approach to texture analysis describes regions using moments of the intensity histogram [148]. Julesz *et al.* [149] showed that the texture of a region cannot be characterised solely by first-order statistics. The spatial greylevel co-occurrence matrix (GLCM) analyses the texture based on the second-order statistics and it has become one of the most well-known and widely used measures [144]. The $(i, j)^{th}$ entry in the matrix $P_{\mathbf{d}}$ is number of times a pair of pixels in an image with grey levels of i and j appear that are separated by a distance **d**. It reveals information about the spatial distribution of the grey levels. For coarse textures the distribution changes by small amounts with distance, whereas fine textures produce larger changes with distance [147]. It must be evaluated for many different vectors **d** for a complete description, thus producing a lot of data. This problem has been addressed by Tou and Chang, who used an eigenvector approach to reduce the feature space [150]. From the GLCM various measures can be found, including entropy, contrast, correlation and homogeneity [147].

The autocorrelation function of an image reveals information about the regularity of the texture and its fineness or coarseness [144]. The autocorrelation function will decrease slowly for a coarse texture and quickly for a fine texture. If the texture primitives are spatially periodic then the autocorrelation function will show oscillations [147].

The spectral power density function assumes the texture primitives are sine and cosine waves. If the 2-D power spectrum is transformed to polar coordinates (r, ϕ) then a peak in the angle ϕ indicates the direction of the texture and a peak in the radius r reveals that the texture has a blob-like constituency [147].

Edgeness per unit area was devised as a measure of the fineness or coarseness of a texture, depending on whether the texture has many or a few edges in a given area [147].

Geometrical Methods

A texture can be considered to be composed of texture elements, or *texels* as they are sometimes known, which are a fundamental micro-structure component. Once the texture element has been identified then either the statistical properties of the placement or deterministic placement rules can be used for analysis [144].

Model-Based Methods

A model can be constructed of a homogeneous texture and then the parameters found for a given image and if the model of a texture is known, then it can be synthesised too. The autoregressive moving averager (ARMA) model seeks to filter noise with an infinite impulse response (IIR) filter to match the texture. The optimum filter coefficients produced are then used for texture analysis. A fine texture produces coefficients that vary widely, whereas the coefficients in a coarse texture are similar [147].

Markov random fields (MRFs) are based on the assumption that the intensity of a pixel is based on that of the surrounding pixels and they have been used extensively for modelling textures [144]. Time series [151] and mosaic models [152] have also been explored.

Many natural objects have a statistical self-similarity at different scales, where an object is composed of smaller copies of itself and a fern is a frequently quoted example. The fractal dimension (FD) gives a measure of the roughness of a surface and the larger the FD, the rougher the surface [144]. For image processing, it is not necessarily the surface roughness that is important, but instead the brightness variations that can be considered on a scale ranging from smooth to rough. Pentland [153] showed that most natural surfaces can be modelled as spatially isotropic fractals. Aerial photographs have been segmented successfully by thresholding the fractal dimension of regions [154].

Signal-Processing Methods

It has been shown that the human visual system transforms the retinal image into a localised space-frequency representation [155]. The same analysis can be performed using the Gabor transform (which is a STFT with a Gaussian window) and wavelet transform techniques [144]. The feature vectors are computed by applying the desired transform and processing the resulting output. In a similar approach, Laws [156] convolved an image region with various kernels and then applied a non-linear operator to determine the textural energy for a given mask.

Conclusion

Having reviewed each of the texture measures, it was decided that the fractal dimension approach would be pursued for the colour mixing research. It was envisaged that natural textures would be used for testing the DFD algorithm, such as wood and rock, for which the fractal dimension is known to be a useful measure. Plastic and metal man-made objects have very little texture, especially if spray-painted, and thus natural textures appear to be more useful for DFD. The FD is also a simple measure in the sense of producing a single parameter, unlike the GLCM, ARMA and MRF approaches.

5.7.3 Introduction to Fractals

Euclid's monumental work *Elements* composed of 13 books and written about 300BC describes the geometry of simple objects through 465 propositions concerning geometry and number theory [157]. Three-dimensional Euclidean geometry is concerned with geometric shapes such as cubes, cones, cylinders and spheres. Observations of the real world reveal shapes that do not approximate these simple primitives as they possess much greater complexity.

Mandelbrot coined the name *fractal* in 1975 [158] and developed a branch of mathematics called *fractal geometry* that is a non-Euclidean type of geometry. The central theme of fractal geometry is that nature exhibits the property of self-similarity. Fractals differ from Euclidean geometrical shapes in that they have a fractional dimension and they are self-similar. Deterministic fractals, such as the Koch snowflake, are generated using well-defined and non-random production rules and random fractals are described statistically [159]. The frond of the fern is a self-similar copy of the whole fern. For surfaces the fractal dimension F_S lies in the range $2 \leq F_S \leq 3$ where $F_S = 2$ implies a smooth surface and $F_S = 3$ means that the surface is very rough. For a volume $3 \leq F_V \leq 4$ and for a Euclidean shape $F_V = 3$. Signals with different fractal dimensions, and thus different roughnesses, are illustrated in Figure 5.3. They were generated using the synthetic power spectrum generation technique.

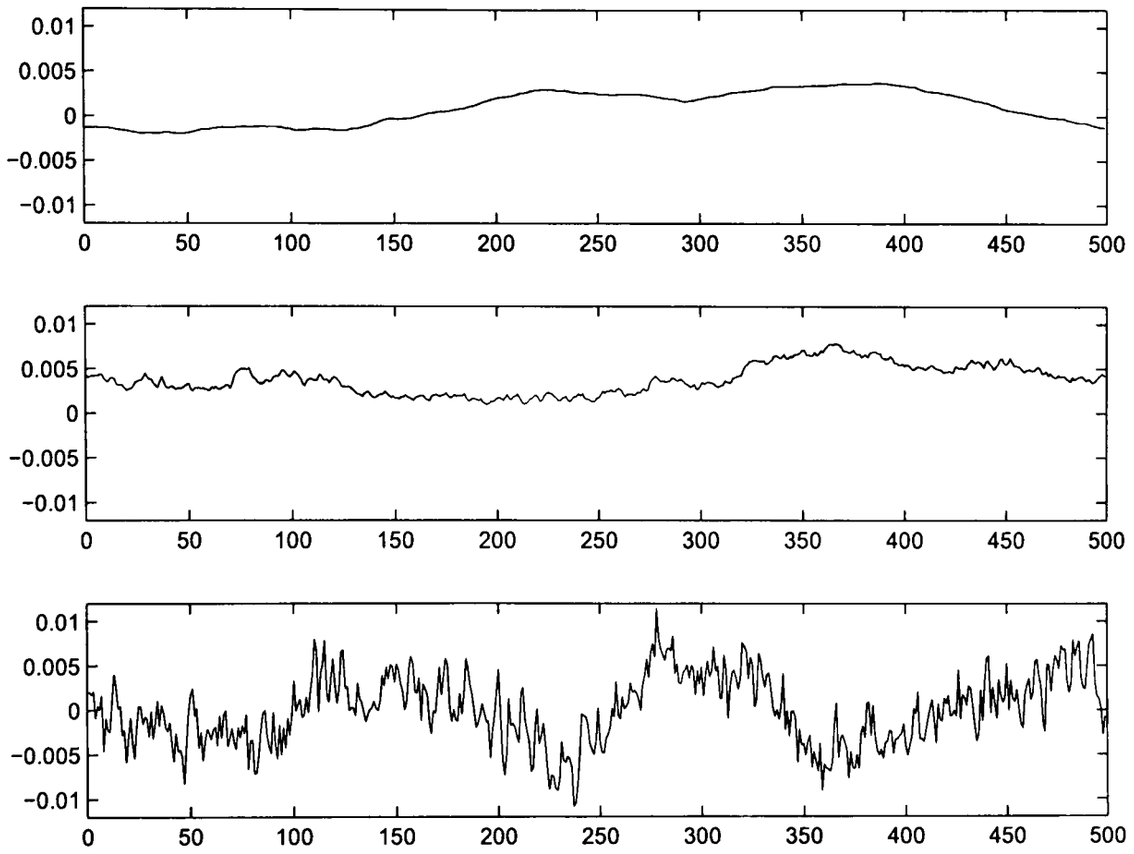


Figure 5.3: Signals with FDs of 1 (top), 1.5 (middle) and 2 (bottom)

Mandelbrot revealed the paradox that the length of a curve depends on the length of the measuring stick and thus the coastline of Britain is infinite in length in the limit [160]. Mandelbrot showed that mountains, clouds and turbulent water have a fractal form [161], but they exhibit self-similarity only [162]. The surfaces of solids are fractals at molecular level and the fractal dimension has been shown to be a way of differentiating and characterising metallic particles in electron microscope images [163]. Fractals have also been used to model rain fall fields [164], interpolate rough curves [165], characterise sea-floor topography [166] and model asteroid surfaces [167] to name just a few.

One of the models of fractals is *fractional Brownian motion* (fBm). An ideal fBm signal has a power spectrum of the form

$$P(k_i) = c\,|k_i|^{-\beta} \tag{5.58}$$

where c is a constant, k_i is the frequency component and β is the spectral exponent that is directly proportional to the Fourier fractal dimension D_F given by

$$D_F = \frac{5-\beta}{2}. \tag{5.59}$$

In image processing fractals have found uses in lossy encoding of images and denoising [168] and further the diffraction properties of fractal apertures are currently being investigated [169].

Whether nature can generally be modelled as a fractal possessing self-similarity to many orders of magnitude is in question, but the power law relationship is clearly useful [170].

5.7.4 Measurement of the Fractal Dimension

There are a multitude of ways to measure the fractal dimension of an image and due to space only a couple can be elucidated here. The box-counting method requires a binary image, which is often the result of thresholding a monochrome image, with a grid layed over the top. The number of squares covered by the black parts of the image are counted. The grid size is then reduced and the process starts again [171].

In order to approximate an image as a fBm model the two spatial coordinates (x, y) are collapsed to a radial component. Assuming the fBm model using Equation (5.58) and an image with a power spectrum $P(k_i)$ the constants β and c can be found using a least-squares fit given by

$$\beta = \frac{N\sum_{i=1}^{N}(\ln P_i)(\ln|k_i|) - \left(\sum_{i=1}^{N}\ln|k_i|\right)\left(\sum_{i=1}^{N}\ln P_i\right)}{\left(\sum_{i=1}^{N}\ln|k_i|\right)^2 - N\sum_{i=1}^{N}(\ln|k_i|)^2} \tag{5.60}$$

and

$$c = \frac{1}{N}\sum_{i=1}^{N}\ln P_i + \frac{\beta}{N}\sum_{i=1}^{N}\ln|k_i| \tag{5.61}$$

where it is assumed that $P_i > 0 \,\forall\, i$ and $k_i > 0 \,\forall\, i$ and N is the number of elements in P_i [161]. The Fourier fractal dimension is then given by substituting the result of (5.60) into (5.59).

Unfortunately, it was not discovered until the end of the research that the least squares fitting method is unstable in the presence of noise. Power and Tullis [172] reported that the higher frequency components are over-represented relative to the lower frequencies in the log-log plots [173]. Noise is more prominent than texture at the higher spatial features, especially with defocused images, thus exasperating the problem of over-representation. Dubuc *et al.* [174] stated that log-log plots rarely produce straight lines, thus increasing the instability in the least-squares fitting, and further the finite number of points makes it difficult to achieve a good fit.

5.7.5 Maximisation of the Fractal Dimension

A textureless surface will have a constant intensity and thus a fractal dimension of $F_S = 2$ and with increasing roughness, the fractal dimension will increase towards $F_S = 3$. Depth-from-defocus algorithms require images that possess texture or appear 'rough' in terms of intensity variations. With a monochrome image the fractal dimension is fixed, but with an RGB colour image the fractal dimension can be changed by scaling the colour planes before addition. Thus, the problem becomes that of finding (α, β, γ) to maximise the fractal dimension, i.e.

$$\max_{(\alpha,\beta,\gamma)} \mathrm{FD}[\alpha R(x, y) + \beta G(x, y) + \gamma B(x, y)] \tag{5.62}$$

where $\mathrm{FD}[\cdot]$ is a function to measure the fractal dimension of an image.

There are more advanced techniques for measuring the fractal dimension assuming a 2-D fBm model [175], or using a fractal interpolation function [176], but the simple model is sufficient for giving a measure of the surface roughness.

5.7.6 Conclusion

The Fourier fractal dimension based on fractional Brownian motion is proposed as a measure of the roughness of the brightness variations of a texture. DFD algorithms rely on the presence of sufficient brightness variations from which to infer the level of defocus. A monochrome image has a fixed texture, but altering the colour planes allows the texture to be changed. It was hypothesised that maximising the fractal dimension through colour mixing would lead to improvements in the depth map.

5.8 Localisation Through Colour Mixing

5.8.1 Introduction

Depth-from-defocus is an ill-posed problem due to noise, undersampling and degradations in the optics of a camera. This section examines a measure of the ill-posedness through a measure known as the *condition number* of a matrix. The preliminary mathematics are discussed before showing how the condition number relates to the error in determining the convolution ratio in Ens and Lawrence's algorithm. The analysis then proceeds to show a particularly interesting feature of the optimum monochrome image.

A passive depth-from-defocus system relies on the scene possessing sufficient texture to accurately calculate the depth map and further the texture is unlikely to be known *a priori* unless the scene is tightly controlled. A data or slide projector can be used to paint the required texture onto a scene using photons. The use of a projected pattern has been employed for DFD before as Pentland *et al.* [76] and Ghita and Whelan [106] [107] projected alternating white and black stripes and Watanabe and Nayar [12] used a checker-board. Thus, monochrome patterns have been developed and this section examines whether there is an advantage to projecting a colour pattern.

5.8.2 Preliminary Mathematics

The definition of a *norm* $\|\cdot\|$ is a function that maps a complex vector space X to a real number, i.e. $\|\cdot\| : X \rightarrow \mathbb{R}$, and it has the properties [177]

1. Positivity: $\|\phi\| \geq 0$
2. Definiteness: $\|\phi\| = 0$ iff $\phi = 0$
3. Homogeneity: $\|\alpha\,\phi\| = |\alpha|\,\|\phi\|$ where α is a constant and $|\cdot|$ denotes modulus
4. Triangle inequality: $\|\phi + \psi\| \leq \|\phi\| + \|\psi\|$

The condition number of a matrix $\mathbf{A}$ is given by

$$C(\mathbf{A}) = \|\mathbf{A}\| \left\|\mathbf{A}^{-1}\right\| \tag{5.63}$$

where $\mathbf{A}^{-1}$ denotes the inverse of matrix $\mathbf{A}$ and the Holder matrix norm of matrix $\mathbf{A}$ is given by

$$\|\mathbf{A}\|_p = \left(\sum_{i,j} |a_{ij}|^p \right)^{\frac{1}{p}}, \quad 1 \leq p \leq 2 \tag{5.64}$$

where a_{ij} is the element of matrix $\mathbf{A}$ in the i^{th} row and j^{th} column. The Euclidean norm is equivalent to the Holder norm with a value $p = 2$, i.e.

$$\|\mathbf{A}\|_2 = \left(\sum_{i,j} |a_{ij}|^2 \right)^{\frac{1}{2}} \tag{5.65}$$

and it can be written as

$$\|\mathbf{A}\|_2 = \sqrt{\mathrm{Tr}\left(\mathbf{A}^\dagger\,\mathbf{A}\right)} \tag{5.66}$$

where $\mathbf{A}^\dagger$ denotes the conjugate transpose of $\mathbf{A}$ and $\mathrm{Tr}(\cdot)$ is the trace of a matrix (i.e. the sum of the leading diagonal entries).

5.8.3 Condition Number Related to Depth-From-Defocus

Introduction

In the description of Ens and Lawrence's algorithm [58] [59] in Section 5.2 the best convolution ratio was determined using a lookup table of convolution ratios and an error measure. A different formulation was also given in their work using matrices where the convolution $i_1(x, y) * h_3(x, y) = i_2(x, y)$ is written as

$$\mathbf{I}_{1\,BT}\,\mathbf{h}_{3\,S} = \mathbf{i}_{2\,S} \tag{5.67}$$

where $\mathbf{I}_{1\,BT}$ is the block-Toeplitz form of image 1, $\mathbf{h}_{3\,S}$ is the row-stacked version of $h_3(x, y)$ and $\mathbf{i}_{2\,S}$ is the row-stacked version of $i_2(x, y)$. A Toeplitz matrix is a diagonal constant matrix and a block-Toeplitz matrix is a matrix composed of Toeplitz sub-matrices. In the noise-free case $\mathbf{h}_{3\,S}$ can be obtained using

$$\mathbf{h}_{3\,S} = \mathbf{I}_{1\,BT}^{-1}\,\mathbf{i}_{2\,S}. \tag{5.68}$$

Therefore, using (5.68) the convolution ratio $h_3(x, y)$ is expressed as a stacked vector $\mathbf{h}_{3\,S}$, but a matrix inverse of the block-Toeplitz version of image 1 is required.

Ill-Posed Problem

The requirement of a matrix inverse in (5.68) makes the problem ill-posed as small changes in the pixel intensities can lead to large changes in the inverse, and thus the resulting convolution ratio, and hence the calculated depth.

Now consider the effect of perturbations in images 1 and 2, denoted $\delta\mathbf{I}_{1\,BT}$ and $\delta\mathbf{i}_{2\,S}$, so that

$$(\mathbf{I}_{1\,BT} + \varepsilon\,\delta\mathbf{I}_{1\,BT})\,\mathbf{h}_{3\,S} = (\mathbf{i}_{2\,S} + \varepsilon\,\delta\mathbf{i}_{2\,S}). \tag{5.69}$$

It can be shown that the relative error in the convolution ratio is given by

$$\frac{\|\delta\mathbf{h}_{3\,S}\|}{\|\mathbf{h}_{3\,S}\|} \le C(\mathbf{I}_{1\,BT})\,\varepsilon\left(\frac{\|\delta\mathbf{I}_{1\,BT}\|}{\|\mathbf{I}_{1\,BT}\|} + \frac{\|\delta\mathbf{i}_{2\,S}\|}{\|\mathbf{i}_{2\,S}\|}\right) + O(\varepsilon^2) \tag{5.70}$$

and thus it can be seen that the condition number of the block-Toeplitz matrix form of image 1 $C(\mathbf{I}_{1\,BT})$ is related to an upper bound on the relative error in the convolution ratio. The condition number of a matrix is a measure of its sensitivity or stability in the presence of small fluctuations. If $C(\mathbf{I}_{1\,BT}) \approx 1$ then the system is well-conditioned, but if $C(\mathbf{I}_{1\,BT}) \gg 1$ then it is ill-conditioned [178].

If a monochrome image $\mathbf{I}_{1\,BT}$ is captured and then scaled by a constant λ it will yield the image $\lambda\,\mathbf{I}_{1\,BT}$ and the condition number of the image is

$$C(\lambda \mathbf{I}_{1\,BT}) = \|\lambda \mathbf{I}_{1\,BT}\| \left\|(\lambda \mathbf{I}_{1\,BT})^{-1}\right\| \tag{5.71}$$

and as

$$(\lambda \mathbf{I}_{1\,BT})^{-1} = \frac{1}{\lambda} \mathbf{I}_{1\,BT}^{-1} \tag{5.72}$$

then

$$C(\lambda \mathbf{I}_{1\,BT}) = \|\lambda \mathbf{I}_{1\,BT}\| \left\| \frac{1}{\lambda} \mathbf{I}_{1\,BT}^{-1} \right\|. \tag{5.73}$$

The homogeneity property of the norm given in Section 5.8.2 means that

$$C(\lambda \mathbf{I}_{1\,BT}) = |\lambda| \, \|\mathbf{I}_{1\,BT}\| \left| \frac{1}{\lambda} \right| \left\| \mathbf{I}_{1\,BT}^{-1} \right\| = C(\mathbf{I}_{1\,BT}) \tag{5.74}$$

and thus shows that scaling a monochrome image will not lead to an improvement in the condition number.

Suppose instead an RGB colour image is captured then the block-Toeplitz forms of the colour planes are denoted $\mathbf{I}_{1\,BT_R}$, $\mathbf{I}_{1\,BT_G}$ and $\mathbf{I}_{1\,BT_B}$ for the red, green and blue planes respectively. A monochrome image is formed from a weighted combination of the colour planes to give

$$\mathbf{I}_{1\,BT} = \alpha \mathbf{I}_{1\,BT_R} + \beta \mathbf{I}_{1\,BT_G} + \gamma \mathbf{I}_{1\,BT_B} \tag{5.75}$$

then it is expected that changing the weights (α, β, γ) affects the condition number of the colour image. The analysis on scaling a monochrome image by λ shows that the condition number is not dependent on the absolute values of (α, β, γ) but on their relative values. The problems encountered with the ill-posedness will not be significantly reduced by finding (α, β, γ) to

$$\min_{(\alpha,\beta,\gamma)} C(\mathbf{I}_{1\,BT}) \tag{5.76}$$

because the formulation does not account for the noise in the system and windowing effects, as shown by the following analysis.

Analysis of the Effect of Noise on the Condition Number

It was found through simulations that choosing (α, β, γ) to minimise the condition number produced worse results than using an equal weighting of the colour planes. The reason for this can be shown theoretically by considering a colour image where one plane is a noise-free signal (e.g. the red plane) and the other is composed only of noise (e.g. the blue plane). The remaining plane is unnecessary for the analysis.

The condition number of a random matrix whose elements are independent and identically distributed normal distribution random variables has been explored by Demmel [179], Edelman [180] and Chen and Dongarra [181]. However, no literature could be

found on the condition number of a block-Toeplitz matrix that is formed from a random matrix. Further, the elements of the matrix must be rounded to simulate quantisation caused by the ADC in the camera, thus introducing non-linear effects.

A numerical approach was sought and for a given mean and standard deviation of Gaussian noise, denoted $N(\mu, \sigma)$, the condition number of the resulting block-Toeplitz matrix was calculated. The condition number of a 5×5 image matrix was tested 15,000 times with a specific realisation of noise $N(\mu, \sigma)$ and the mean condition number of the matrix is plotted in Figure 5.4. The reason a small image matrix was used is that for an $n \times n$ matrix the corresponding block-Toeplitz matrix is $n^2 \times n^2$ and this matrix must be inverted in order to calculate the condition number. Thus, the time taken to perform the simulation increases rapidly with n.

Where the condition number is not defined in Figure 5.4 it is because at least one of the condition numbers in the test was infinite, i.e. the matrix was singular to working precision.

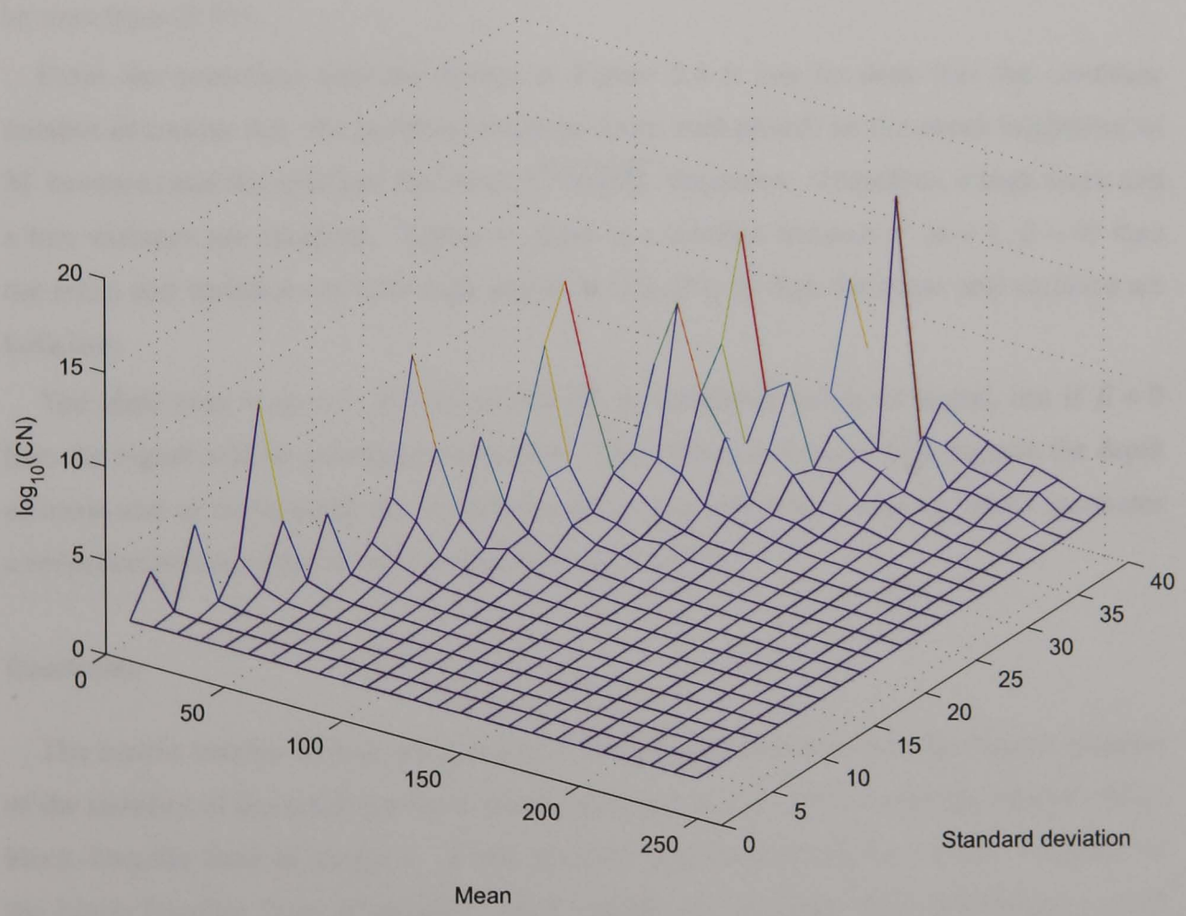


Figure 5.4: The mean condition number as a function of $N(\mu, \sigma)$ for a 5×5 image matrix

Now consider the resulting monochrome image formed from scaling the red and blue colour planes **R** and **B** using

$$\mathbf{M} = \alpha \mathbf{R} + \gamma \mathbf{B} \tag{5.77}$$

where matrix notation has been used for the image and α and γ are the real scaling constants. The mean and variance of monochrome image matrix **M** are given by

$$E[\mathbf{M}] = \alpha E[\mathbf{R}] + \gamma E[\mathbf{B}] \tag{5.78}$$

$$\text{Var}[\mathbf{M}] = \alpha^2 \text{Var}[\mathbf{R}] + \gamma^2 \text{Var}[\mathbf{B}] + 2\alpha\gamma \text{Cov}[\mathbf{R}, \mathbf{B}] \tag{5.79}$$

where $E[\mathbf{X}]$ and $\text{Var}[\mathbf{X}]$ denote the expectation and variance of the elements in matix **X**. The image texture (red plane) will be assumed to have a Gaussian brightness distribution and the noise (blue plane) will be assumed to be AWGN with a mean of zero, i.e. $E[\mathbf{B}] = 0$. Further, it is assumed that the image and noise are independent so that $\text{Cov}[\mathbf{R}, \mathbf{B}] = 0$. Therefore, (5.78) and (5.79) reduce to

$$E[\mathbf{M}] = \alpha E[\mathbf{R}] \tag{5.80}$$

$$\text{Var}[\mathbf{M}] = \alpha^2 \text{Var}[\mathbf{R}] + \gamma^2 \text{Var}[\mathbf{B}]. \tag{5.81}$$

It will also be assumed that $\text{Var}[\mathbf{B}] < \text{Var}[\mathbf{R}]$ so that the SNR is greater than 0dB, as can be seen from (5.41).

From the numerical analysis shown in Figure 5.4 it can be seen that the condition number decreases (i.e. the problem becomes more well-posed) as the mean brightness of **M** increases and the standard deviation $\sqrt{\text{Var}[\mathbf{M}]}$ decreases. Therefore, a high mean and a low variance are required. However, there is a conflict because if ($\alpha = 1, \beta = 0$) then the mean and variance are both high and if ($\alpha = 0, \beta = 1$) then the mean and variance are both low.

The ideal case is ($\alpha = 1, \beta = 0$) so that **M** is composed solely of signal, but if $\beta \neq 0$ then the signal will be corrupted with noise. The presence of noise will degrade the depth estimate and as minimising the condition number through colour mixing cannot guarantee a reduction in noise (due to the conflict) it is not suitable.

Conclusion

The matrix version of Ens and Lawrence's algorithm has been discussed and a measure of the stability of the result has been shown through the use of the condition number of the block-Toeplitz form of image 1. It was assumed that minimising the condition number of the block-Toeplitz form of image 1, thus making the problem more well-posed, would decrease the depth error. However, the analysis has shown that minimising the condition number can lead to a reduction in the SNR and thus an increase in the depth error.

The theoretical analysis has revealed why colour mixing to minimise the condition number is not suitable for DFD as it can have the effect of increasing the noise in the system. However, it was instructive to examine the form of the monochrome image $\mathbf{I}_{1\,BT}$ that gives the minimum condition number and this is the subject of the next section.

5.8.4 Monochrome Image with Minimum Condition Number

A Genetic Algorithm was used to evolve the pixel intensities of a monochrome image such that the image in block-Toeplitz form possessed the lowest possible condition number. With sufficient generations it was found that the image was of the form

$$I = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix}. \tag{5.82}$$

The particularly interesting feature of the image I is that it has good *localisation*, i.e. only one pixel has a non-zero value and so the PSF $\mathbf{h}_{3S}$ that results will only be due to that particular pixel. For that pixel the image overlap problem does not exist as it is not being affected by the intensity content due to surrounding pixels. This analysis lead to the idea of a projected colour pattern where the colours are carefully chosen to maximise the *Localisation through Colour Mixing* (LCM). In the look-up table approach it is not the first pixel that is most important, but instead the centre pixel. Thus, as an example the optimum 5×5 image region would be

$$I = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{5.83}$$

This corresponds to the simplest and slowest method where a white dot or square is projected onto the scene. The scene is imaged and then the depth is calculated using a DFD algorithm. In order to generate a complete depth map the white dot must be moved to the next pixel position and the process repeated. There is essentially no problem with localisation with this method, but it is time consuming. Ideally, a colour pattern needs to be projected onto the scene and then colour mixing used to recover the required components due to the central pixel in a window.

5.8.5 The Pattern Development

The purpose of a projected pattern is firstly to overcome a lack of texture, but also provide a texture that has the required properties. A bright pattern is likely to have a better signal-to-noise ratio than a dark pattern for given camera parameters and so reduce the effect of noise, but this is primarily a practical issue. One of the main problems with Ens and Lawrence's algorithm is that the equifocal assumption was applied in the derivation, where the depth is assumed to be constant within a window. This section considers the development of a projected pattern so that the contribution due to the centre pixel is maximised in relation to the other pixels in the window, thus producing better localisation.

Consider the simple case of three colour pixels where the aim is to find the optimum scaling parameters (α, β, γ) such that the contribution in intensity due one of the pixels can be separated from the other two. For example, suppose one pixel is cyan, one is magenta and the other yellow, i.e. the three secondary colours. The three pixel, RGB image can be expressed in a matrix form as

$$\Psi = \begin{pmatrix} R_1 & G_1 & B_1 \\ R_2 & G_2 & B_2 \\ R_3 & G_3 & B_3 \end{pmatrix} \tag{5.84}$$

where R_i, G_i and B_i are the red, green and blue components of the i^{th} pixel respectively. The matrix Ψ for the example using the secondary colours is given by

$$\Psi = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \tag{5.85}$$

as cyan is composed of green and blue, magenta is composed of red and blue and yellow is a combination of red and green. It has been assumed that equal weightings of the primaries occurred, although this is not necessary.

Let the vector $\mathbf{m}$ denote the required monochrome brightness pattern. If the problem is to extract the content due to the middle pixel (i.e. $i = 2$) then the vector takes the form

$$\mathbf{m} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}. \tag{5.86}$$

The problem then becomes that of finding the optimum scaling vector $\mathbf{s} = (\alpha, \beta, \gamma)^T$ such that

$$\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} \tag{5.87}$$

and so a solution to

$$\Psi \mathbf{s} = \mathbf{m} \tag{5.88}$$

must be found for $\mathbf{s}$. If Ψ is square (i.e. it represents exactly three colour pixels) then the optimum scaling constants (α, β, γ) are found simply from

$$\mathbf{s} = \Psi^{-1} \mathbf{m}. \tag{5.89}$$

For the example,

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{5.90}$$

and it can be shown that $\alpha = \frac{1}{2}, \beta = -\frac{1}{2}, \gamma = \frac{1}{2}$ as

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{2} \\ -\frac{1}{2} \\ \frac{1}{2} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}. \tag{5.91}$$

The inverse of the square matrix Ψ exists if and only if the determinant is not zero. The determinant is zero if two or more rows (and thus columns) are linearly dependent and this occurs if two pixels have the same colour (i.e. hue and saturation) but possibly different brightnesses.

When there are more than three pixels the pseudo-inverse must be employed as the matrix Ψ is not square. The Moore-Penrose matrix inverse Ψ^+ of a matrix Ψ has the desirable property that $\mathbf{s} = \Psi^+ \mathbf{m}$ is the shortest length least squares solution to the problem $\Psi \mathbf{s} = \mathbf{m}$. The Moore-Penrose pseudo-inverse is given by [182]

$$\Psi^+ = \Psi^T \left(\Psi \Psi^T \right)^{-1} \tag{5.92}$$

where Ψ^T represents the transpose of Ψ.

Consider the problem of finding the optimum scaling constants (α, β, γ) to extract the i^{th} pixel from the colour image with N RGB pixels in a least squares sense. It can be written explicitly as

$$\begin{pmatrix} R_1 & G_1 & B_1 \\ R_2 & G_2 & B_2 \\ \vdots & \vdots & \vdots \\ R_{i-1} & G_{i-1} & B_{i-1} \\ R_i & G_i & B_i \\ R_{i+1} & G_{i+1} & B_{i+1} \\ \vdots & \vdots & \vdots \\ R_N & G_N & B_N \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \tag{5.93}$$

where R_i, B_i and G_i are the red, green and blue intensities of the i^{th} pixel respectively. This approach has been named *Localisation through Colour Mixing* (LCM) as it is a method of localising the content of an image segment using colour mixtures.

Nayar *et al.* [91] [12] showed that were two optimum projected patterns for their formulation of DFD: a checkerboard pattern composed of black-and-white squares that were the same size as the pixels and exactly aligned to the CCD; the other being a checkerboard with squares twice as big as the sensor's pixels, but with a phase shift of half a pixel. In order to use the pattern for DFD it must be projected onto a scene such that one pixel of the pattern corresponds to one pixel on the CCD. This can be achieved using a telecentric projector [10] [12] [13].

As a theoretical analysis could not be performed, it was assumed that the coloured pattern would consist of squares that occupy one pixel and an optimisation was used to choose the colours of the squares. The choice of the colour pattern determines how well a particular pixel can be extracted from the others and the next section considers an evolutionary algorithm approach to finding the pattern.

Appendix F highlights the importance of taking depth discontinuities into consideration. When an image window straddles two objects at different depths, the depth returned was found experimentally be that due to the closer object. If the centre of the window is on the closer object then it is not a problem, however, if the situation is reversed, the depth error can be significant. Depth discontinuities in the form of occluding steps were analysed by Asada *et al.* [183] assuming geometrical optics and constant intensity regions. Despite simple assumptions, their work showed that the brightness transition of a blurred occluding edge is the same as would be produced by a surface edge (i.e. a brightness edge) on the occluding object, provided the brightness of the occluding surface is uniform. Thus, the edge appears to be due to the occluding surface and not the occluded surface. Since the step closer to the camera is the occluding step, the depth returned (which is found by examining the brightness transition) is due to that and not the occluded step, i.e. the one further from the camera.

LCM should help to reduce the depth error around step discontinuities by removing some of the intensity contribution due to the occluding or occluded region as required.

5.8.6 The Genetic Algorithm to Optimise the Projected Pattern

Consider the colour image in matrix form and denoted Ψ and composed of N RGB pixels. Suppose the required monochrome image is denoted $\mathbf{m}$ and so the scaling constants (α, β, γ) are given by

$$\mathbf{s} = \Psi^{+}\,\mathbf{m} \tag{5.94}$$

and the actual monochrome image $\mathbf{m}_a$ formed using scaling $\mathbf{s}$ is given by

$$\mathbf{m}_a = \Psi\,\mathbf{s} = \Psi\,\Psi^{+}\,\mathbf{m} \tag{5.95}$$

and generally $\mathbf{m}_a \neq \mathbf{m}$. The mean squared error for the particular pattern is given by

$$\varepsilon = \frac{1}{N}\,(\mathbf{m} - \mathbf{m}_a)\,(\mathbf{m} - \mathbf{m}_a)^T. \tag{5.96}$$

Now suppose the vector $\mathbf{m}_i$ is composed of all zeros except for an entry of value one. For test i the unit entry would appear at index i. For example, if the image is composed of six pixels and $i = 2$ then

$$\mathbf{m}_2 = (0,\ 1,\ 0,\ 0,\ 0,\ 0). \tag{5.97}$$

The overall mean squared error E for testing all N combinations is given by

$$E = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{m}_i - \mathbf{m}_{i_a})\,(\mathbf{m}_i - \mathbf{m}_{i_a})^T \tag{5.98}$$

where $\mathbf{m}_{i_a}$ is the actual monochrome image.

The resolution of a typical image employed for DFD is 640×480 giving a total of 307,200 colour pixels and clearly it would be quite a task to ensure the MSE is approximately the same for all pixels. The implementation of Ens and Lawrence's algorithm employs a 32×32 window that is moved in the required increments in the x and y directions. By tiling the pattern as shown in Figure 5.5 it can be seen that the same colour pixels are presented to the window, but not necessarily in the same order. Thus, the problem reduces to finding a total of 1024 colour pixels for a 32×32 window. In the example presented in Figure 5.5 with an image size of 6×6 and a window size of 3×3 only 9 different colours are required.

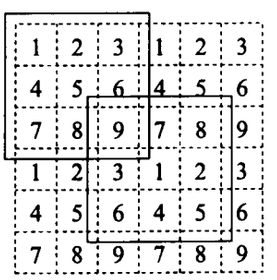


Figure 5.5: The 3×3 tiled pattern where each number represents a distinct colour

A Genetic Algorithm was written to find an optimum pattern composed of 1024 pixels such that each pixel could be extracted from the others by applying colour mixing using the pseudo-inverse in a least squares sense. It was noticeable that the objective value changed little from one generation to the next, suggesting that the random pattern at the start of the evolution was near optimum.

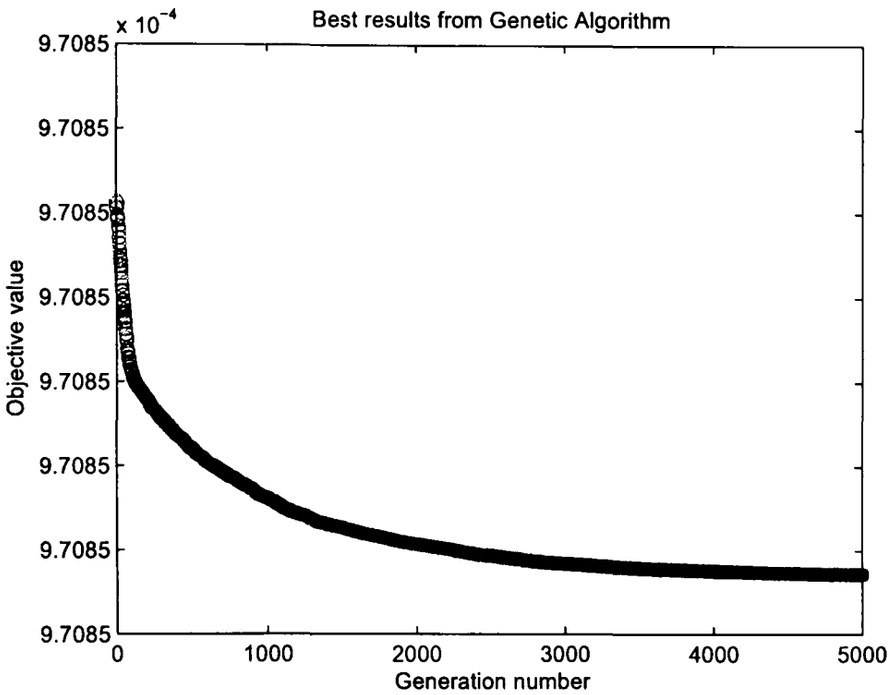


Figure 5.6: Objective value (the MSE) as a function of the generation number

The reason that there was only a limited improvements in the optimisation was believed to be due to the fact that once the matrix Ψ reached full-rank, which in this case is 3, the last $N-3$ rows are linearly dependent on the first 3 rows. Thus, the optimum pattern is a randomly coloured checkerboard pattern.

Watanabe and Nayar [81] used a 7×7 window and Favaro and Soatto [94] used 7×7 and 9×9 and with fewer colour pixels to choose the MSE is lower.

5.8.7 Conclusion

The optimum monochrome image from the perspective of analysis based on the condition number showed that the image has very good localisation properties. Localisation reduces the edge effect and the image overlap problem and thus should help to produce more accurate depth maps. The projected pattern evolved using the GA ensures that there is sufficient texture present and the colours in the texture were chosen to maximise the localisation properties. Note that the assumption was that the colour pattern would be projected onto white (or grey) surfaces. Coloured surfaces would pose a problem because they only reflect their own colour.

5.9 Conclusion

A derivation of Ens and Lawrence's algorithm based on searching for the depth using a look-up table of convolution ratios has been discussed. Only in noise-free simulations will an entry in the look-up table give $i_1(x, y) * h_3(x, y) = i_2(x, y)$ and so an error measure must be employed. Ens and Lawrence chose to use the sum of the L_2-norms and Chapter 6 presents results using the total-variation and the I-divergence measures.

The concept of forming linear combinations of the red, green and blue colour planes captured by a colour camera was presented and shown to be an approximation to using a physical colour filter, but with the advantage that the filter does not need to be known *a priori*. The basis of the research from the initial Genetic Algorithm idea was discussed and an example to show the merit of colour mixing given.

The use of the GA with a known depth is not a practical solution, but merely a research tool. Deterministic algorithms were presented where each was designed with a particular DFD problem in mind. PCA is a standard technique to use on colour images and it has the desired property of producing decorrelated image planes with decreasing levels of variance. The maximisation of the fractal dimension was designed to ensure sufficient texture. The maximisation of the SNR was designed to reduce the effect of noise and it was based on an additive noise model.

The minimisation of the condition number of the block-Toeplitz matrix form of image 1 was presented for academic interest, but clearly the approach is not expedient for practical colour mixing for DFD. However, it seeded the idea of colour mixing with a projected pattern. The Localisation through Colour Mixing (LCM) algorithm was designed to reduce the windowing effect and thus should perform better than the monochrome case when the object is very defocused or there is a depth discontinuity.

Table 5.1. Summary of algorithms developed

Texture	Signal-to-Noise Ratio	Windowing effect
PCA	Maximisation of SNR	Localisation through Colour Mixing
Maximisation of Fractal Dimension		

Chapter 6

The Results of Colour Depth-From-Defocus

6.1 Introduction

The main goal of most non-volumetric 3D imaging systems is to produce depth maps that give an estimate of the depth of points in the scene from the camera. The more accurate and reliable the depth map, the more useful the system is for its intended application. Previous work on DFD was based on monochrome images and Chapter 5 presented possible ways to improve the depth maps by using a colour camera and a linear combination of colour planes to give an optimum monochrome image. This chapter examines their effectiveness as a pre-processing stage for Ens and Lawrence's [58] [59] depth-from-defocus algorithm. There are many variables in a DFD system including:

- Camera parameters used (e.g. the two different f-numbers employed)
- Distance between the camera and the object
- Orientation of the object relative to the optical axis
- Textural and colour properties of the object
- Noise properties of the camera

One of the problems of empirically testing the algorithms is that it is very difficult to conclusively show their effectiveness as it depends on all of the factors listed above and so simulations were performed to ensure tightly controlled experiments. Ens and Lawrence's algorithm was based on the assumption that the depth is constant within a window and so one of the first tests of the algorithms was with planes perpendicular to the camera.

In Section 6.2 the MATLAB implementation of the DFD algorithm and the pre-processing stage are discussed. The image window size and the size of the convolution ratio are examined through experiments using a checkerboard. The initial results of the research using a Genetic Algorithm that evolves the optimum scaling constants (α, β, γ) for a

known depth map are presented in Section 6.3 along with an analysis in the presence of noise. The next sections each consider a pre-processing algorithm developed in turn. Principal Component Analysis (PCA) was employed and the results are given in Section 6.4. The presence of noise is clearly inevitable in any real imaging system and Section 6.5 presents results of using colour mixing to maximise the signal-to-noise ratio (SNR). The fractal dimension (FD) of a texture gives a measure of its roughness and an optimisation algorithm was written to maximise the FD using colour mixing, before the resultant monochrome image was applied to the DFD algorithm. The results are given in Section 6.6. The *Localisation through Colour Mixing* (LCM) was designed to alleviate some of the problems caused by windowing and the image overlap problem and the simulation results are examined in Section 6.7. Finally in Section 6.8 the results are summarised and conclusions drawn.

6.2 Implementation and Initialisation

6.2.1 Introduction

This section discusses the implementation of the DFD algorithm and the tests that were performed before the colour mixing aspect could be examined.

6.2.2 Software Implementation

The DFD algorithm was implemented in MATLAB as it provides many useful digital image processing functions and although it ran slower than an implementation would in C, it made debugging and prototyping faster and easier. A lot of the optimisation algorithms were based on a Genetic Algorithm using the Genetic Algorithm Toolbox for MATLAB written at the University of Sheffield.

6.2.3 Simulation of Defocused Images

The point spread function data for the 24mm Sigma photographic lens was vital for the generation of the required convolution ratios, but it also meant that simulated defocused images could be created. The impetus for doing simulations was that the experiments could be very carefully controlled and the depth maps known exactly. A focused scene $f(x, y)$ can be defocus blurred using a spatially-varying kernel $h(x, y, \xi, \eta)$ to give a defocused image $i_k(x, y)$ using

$$i_k(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi, \eta)\, h(x, y, \xi, \eta)\, d\xi\, d\eta \tag{6.1}$$

where $h(x, y, \xi, \eta)$ represents the blurring at position (x, y) as a result of the point (ξ, η). The discrete approximation is given by

$$i_k(x, y) = \sum_{\xi=0}^{M-1}\sum_{\eta=0}^{N-1} f(\xi, \eta)\, h(x, y, \xi, \eta) \tag{6.2}$$

where the image is of size $M \times N$. Equation (6.2) was implemented in MATLAB to produce simulated defocused images with only quantisation noise present.

6.2.4 Error Measurements and Optimum Window Size

In order to calculate the depth of a point in the scene two images are taken and then a window applied centred on the point of interest. The size of the window is important because if it is too small there is insufficient information and if it is too large the depth returned will be dependent on the surrounding areas too (and this is discussed in Section 6.2.11). A similar consideration concerns the convolution ratio lookup table that consists of pre-computed Gaussian functions with the required standard deviations. Near the focus position the Gaussian PSFs have a small spread and consequently so does the convolution ratio as shown in Figure 6.1. As the distance increases the relative spread of the PSFs increases and so does the support of the convolution ratio. If the support remains the same then the measure

$$\varepsilon = \sum_{x,y} (i_1(x, y) * h_3(x, y) - i_2(x, y))^2 = \sum_{x,y} \left(\hat{i}_2(x, y) - i_2(x, y)\right)^2 \tag{6.3}$$

can be used, which is that used by Ens and Lawrence. If the support of the Gaussian convolution ratio kernel changes with size then the error measure must be normalised to give

$$\varepsilon = \frac{1}{N}\sum_{x,y} \left(\hat{i}_2(x, y) - i_2(x, y)\right)^2 \tag{6.4}$$

where N is the number of elements in $i_2(x, y)$ and thus $\hat{i}_2(x, y)$ too.

6.2.5 Generation of the Convolution Ratios

The accuracy of the depth maps are highly dependent on the precision of the PSFs. Chapters 3 presented improvements in finding the PSFs and results of using the 16mm video lens and the 24mm Sigma lens were given. Due to the presence of aberrations in the video lens, it was decided that the 24mm lens would be used for the DFD experiments. Out of all the PSF shapes tested, the step in intensity with non-uniform illumination and a Generalised Gaussian model of the PSF fitted the ESFs better, but the problem then came to accurately determine the convolution ratios.

No closed form solution could be found that linked the Generalised Gaussian PSFs of cameras 1 and 2, denoted $h_1(x, y)$ and $h_2(x, y)$, to the convolution ratio $h_3(x, y)$ such that

$$h_1(x, y) * h_3(x, y) = h_2(x, y) \tag{6.5}$$

and so a GA was written to evolve the convolution ratio. Unfortunately, the results were not consistent and coupled with the obvious lack of precision in determining the power of the Generalised Gaussian it was decided that the simpler Gaussian model would be used. The relation between the standard deviations of the PSFs for cameras 1 and 2 and the convolution ratio $h_3(x, y)$ is discussed in Appendix D. Thus, in all the DFD results presented in this and the following chapter, Gaussian PSFs have been assumed and not the Generalised Gaussian PSFs.

The two defocused images required for DFD must be captured with different intrinsic camera parameters, which are the f-number, focal length and the focus position (i.e. the lens to CCD distance), as discussed in Section 1.1.2. DFD algorithms are reported to be less sensitive to depth variations when changing the aperture alone compared to changing the focus position, however, it avoids magnification effects [74]. Further, the lens was moved manually and it was easier to consistently set the f-number compared to changing the focus position.

The standard deviation of the Gaussian convolution ratio for the 24mm Sigma photographic lens for three aperture combinations are shown in Figure 6.1. The DFD system required two images to be taken with two different apertures and they are referred to as f_1 and f_2 respectively. The smaller the aperture (and thus the larger the f-number) the more focused the image. The algorithm developed by Ens and Lawrence requires $f_1 > f_2$ and if $f_1 = f_2$ then there is no relative defocus and so no depth information.

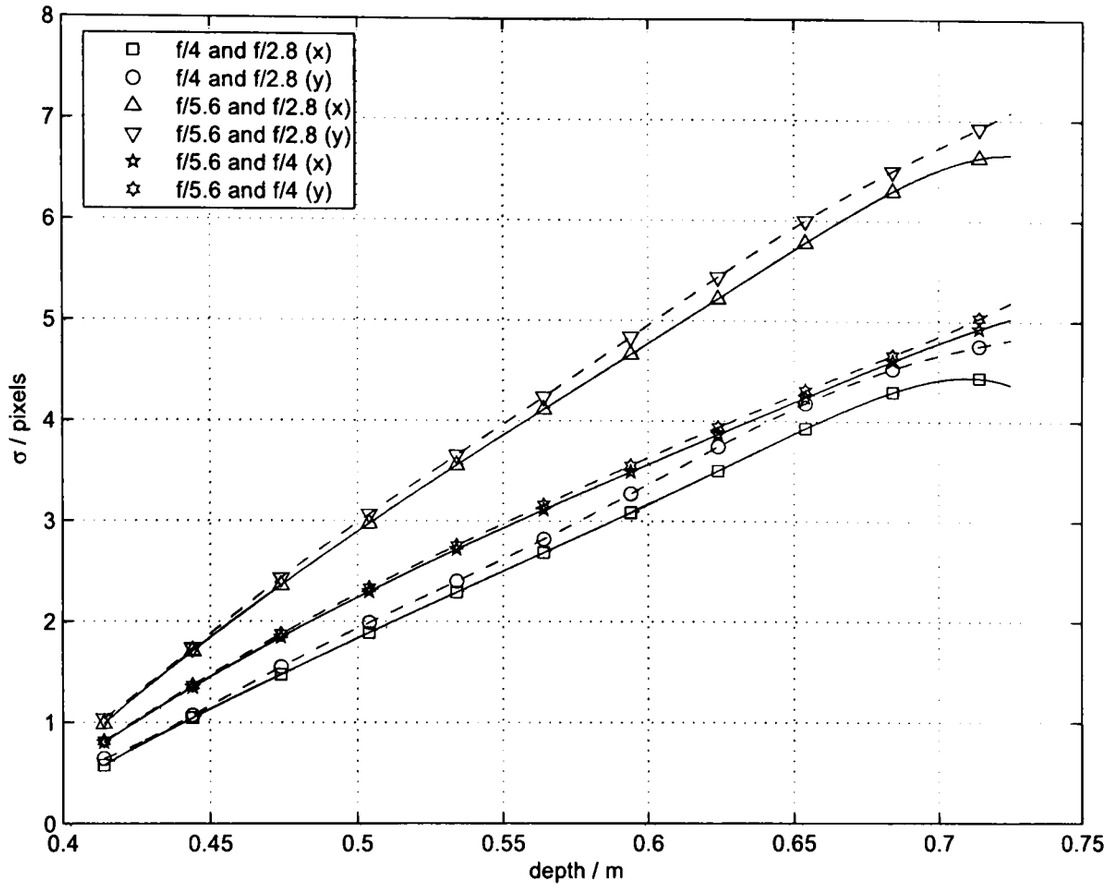


Figure 6.1: The convolution ratios for the 24mm Sigma photographic lens

The standard deviations of the Gaussian convolution ratios are fairly linear with depth, but the maximum object range must be limited to about 0.7m due to the non-monotonic nature using aperture combinations of ($f_1 = 4$, $f_2 = 2.8$). The aperture combination ($f_1 = 5.6$, $f_2 = 2.8$) has the largest standard deviation because it has the biggest difference in aperture sizes.

There are slight differences in the x and y directions as indicated by Figure 6.1, but the effect is much less pronounced than if the video lens was used. The fact that the standard deviations are smooth and that for all aperture combinations the spread is consistently larger in the y direction than the x direction suggests that the effect cannot be attributed to camera noise. The pixels of the CCD in the Basler A631fc colour camera were square and so it was expected that the PSF would be circularly symmetric. Lens imperfections were believed to cause the slight deviations from circular symmetry.

Ens and Lawrence [58] [59] used the sum of the L_2-norm as a measure of the error between the actual defocused image $i_2(x, y)$ and the approximation $\hat{i}_2(x, y)$ formed by blurring the first defocused image using a possible convolution ratio $h_3(x, y)$, given by $\hat{i}_2(x, y) = i_1(x, y) * h_3(x, y)$. As discussed in Section 5.2.2 this is not the only error measure and it was decided that two others would be tested, namely the sum of the L_1-norm, known as total variation and given by

$$\varepsilon = \sum_{x,y} \left| \hat{i}_2(x, y) - i_2(x, y) \right| \tag{6.6}$$

and the Kullbach's information-divergence [133], given by

$$I(\hat{i}_2 \,||\, i_2) = \sum_{x,y} \left(\hat{i}_2(x, y) \log\left(\frac{\hat{i}_2(x, y)}{i_2(x, y)} \right) - \hat{i}_2(x, y) + i_2(x, y) \right). \tag{6.7}$$

Various combinations of the image and convolution ratio window sizes had to be tested as it could not be assumed that they are independent.

6.2.6 Speed Improvement

In Ens and Lawrence's implementation of the lookup table approach every convolution ratio had to be tested to find the one that produces the minimum error. A plot of the error ε versus depth is fairly smooth, as shown in Figure 6.2, but no derivative could be found. In order to improve the speed of the algorithm, 19 equally spaced positions of the possible 314 were tested and then every position within the region between the three that gave the lowest error were tested. This crude improvement reduced the time taken to calculate the depth by a third.

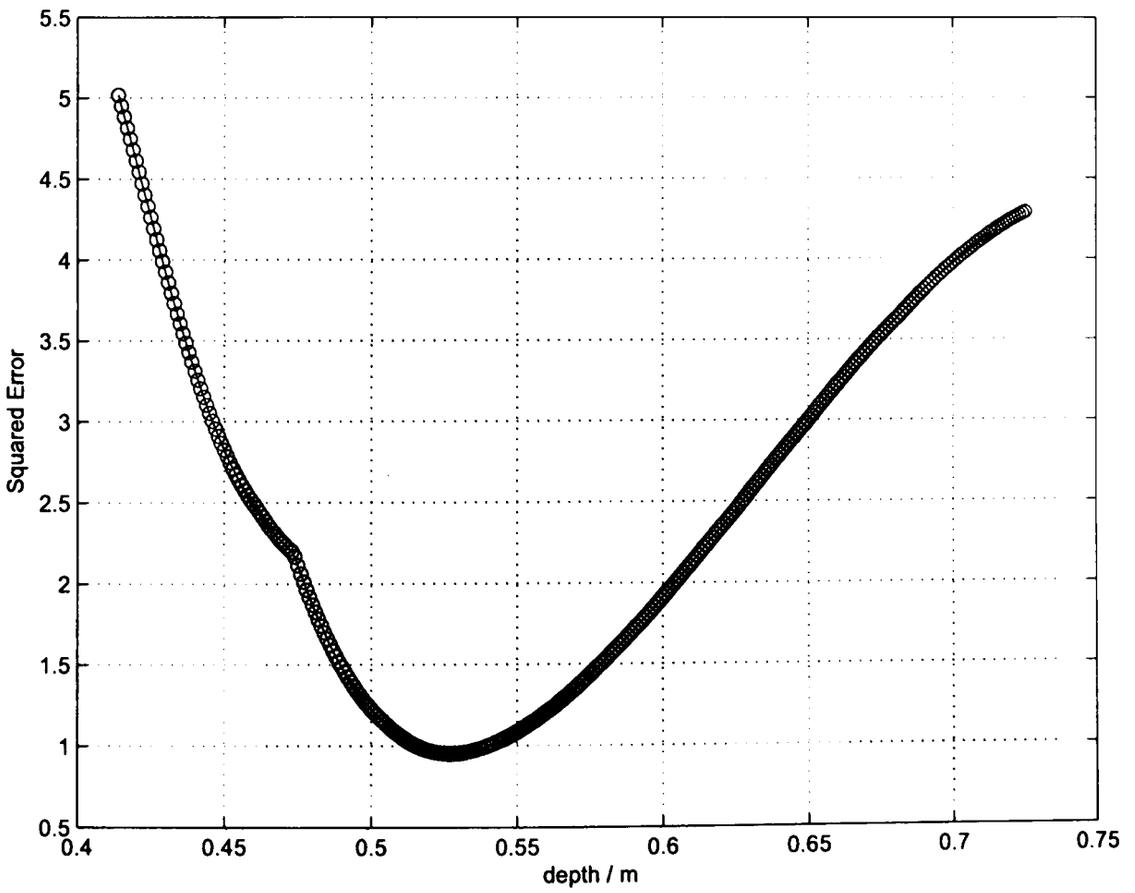


Figure 6.2: The squared error versus depth

Although the execution time was not an issue - as the primary interest of the research was not developing a real-time DFD algorithm, but instead a colour DFD algorithm - the speed improvement meant that three times the number of simulations could be performed in a given time.

6.2.7 Post-Processing Algorithm

The block shift-variant algorithm devised by Rajagopalan and Chaudhuri [92] imposed restrictions on the depth map to ensure smoothness whereas Pentland's [76] algorithm used wavelet regularisation as a post-processing step. The formulation of Ens and Lawrence's [59] algorithm requires a post-processing step to reduce noise in the depth map and so a moving 3×3 median filter window was employed. The small kernel ensures the depth map is not excessively smoothed.

6.2.8 Depth Map Error Measures

In comparing the depth maps it is instructive to have some statistics that can be used for analysis. The error $\varepsilon(x, y)$ in the depth map is defined as

$$\varepsilon(x, y) = z(x, y) - \hat{z}(x, y) \tag{6.8}$$

where $z(x, y)$ is the depth map produced by the DFD algorithm and $\hat{z}(x, y)$ is the actual depth map, both of which are of size $M \times N$. The mean depth error $\overline{\varepsilon}$ is then given by

$$\overline{\varepsilon} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \varepsilon(x, y) \tag{6.9}$$

and if, for example, the mean depth is positive then the DFD algorithm has over-estimated the depth on average. The variance of the error is given by

$$\sigma^2 = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (\varepsilon(x, y) - \overline{\varepsilon})^2 \tag{6.10}$$

and it gives a measure of the spread of the error. Often in DFD papers the Mean Square Error (MSE) is quoted and it is a measure of the goodness of fit of the depth map $z(x, y)$ produced using DFD to the actual, known depth map $\hat{z}(x, y)$. It is defined as

$$\text{MSE} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \varepsilon^2(x, y). \tag{6.11}$$

6.2.9 Number of images required for averaging

A random coloured texture was pasted to a slope that had a minimum distance of 0.440m and a maximum distance of 0.524m from the camera. One hundred and twenty-eight images were taken of the scene for a given aperture setting and then the depth error using the PCA and monochrome algorithms was calculated where 1, 2, 4, 8, 16, 32, 64 and 128 images were averaged. The MSE followed the $\sqrt{N}/N$ model fairly well as expected, as shown in Figure 6.3 where the apertures employed were f/5.6 and f/2.8.

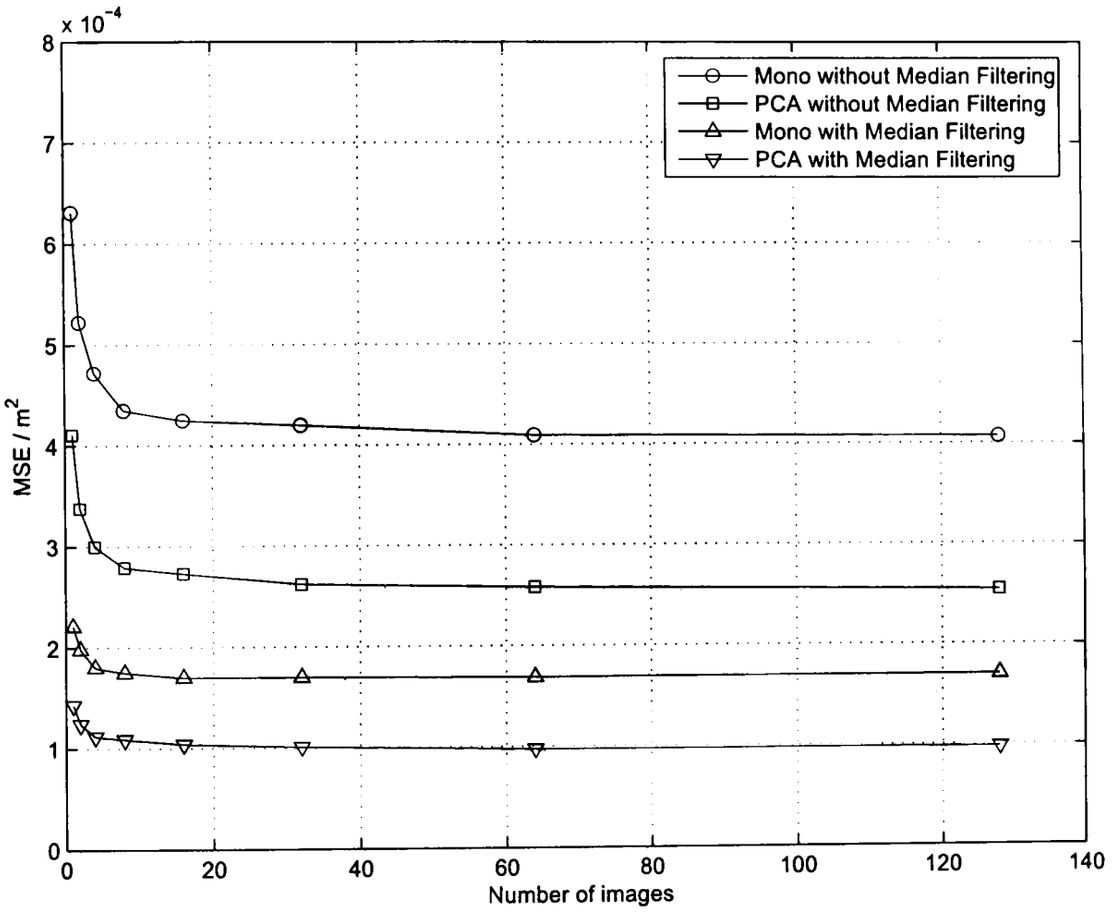


Figure 6.3: MSE as a function of the number of images averaged using f/5.6 and f/2.8

The law of diminishing returns is clearly active in the averaging and so only about 8 images need to averaged in practice. The median filtering has clearly reduced the MSE for the monochrome and PCA algorithms and this is discussed in more details in the later sections.

6.2.10 Checkerboard Results

Images of a checkerboard pattern perpendicular to the camera's optical axis were obtained using the Basler A631fc colour camera and the 24mm Sigma photographic lens for a range of depths with three different apertures: f/2.8; f/4; and f/5.6. The images were processed using the MATLAB-based implementation developed and results presented in Appendix D show the mean and standard deviation of the recovered depth as a function of the width of the image window and the width of the convolution ratio window for the three different error measures discussed, namely the sum of the L_1 - and L_2 -norms and the Information-Divergence. It was found that the L_2 -norm, as used by Ens and Lawrence [58] [59], performed better than both the L_1 -norm and the I-Divergence, where the latter was particularly sensitive to the relative scaling between the images and noise.

It was found that a 64×64 image window produced depth maps that were smoother, i.e. had a lower variance of depth error, but with the consequence of poorer localisation and longer processing times. For a 32×32 image window, a fixed convolution ratio window size of 21×21 produced much better results than allowing the convolution ratio to vary.

Of the three different possible aperture combinations, using ($f_1 = 4$, $f_2 = 2.8$) produced much worse results. This was believed to be due to the significant blurring in both images causing a severe reduction in the information content and increasing the image overlap problem.

6.2.11 Localisation Analysis

Introduction

If the defocused images are not windowed, i.e. the entire of each image is used, then the DFD algorithm will return a single depth estimate. If the image is windowed to give four non-overlapping regions then four depth estimates will be found, thus increasing the depth localisation. A single pixel of the image in contrast does not give any defocus information. Thus, there is an optimum window size somewhere between the two extremes. This section examines the effect of window size on the depth localisation.

Watanabe and Nayar [81] designed their algorithm to have as small a window size as possible for their algorithm because they knew a small kernel size leads to a depth map with a high spatial resolution. However, they recognised that the uncertainty principle meant that the frequency resolution decreases proportional to the inverse kernel size used.

Experiments

A randomly coloured checkerboard pattern with 5×5 pixel squares was texture mapped onto a steps scene with 10 steps equally spanning the depth range 0.42m to 0.62m. The actual depth map is shown in Figure 6.4. The scene was then defocus blurred to simulate being taken by a colour camera with apertures of f/5.6 and f/2.8.

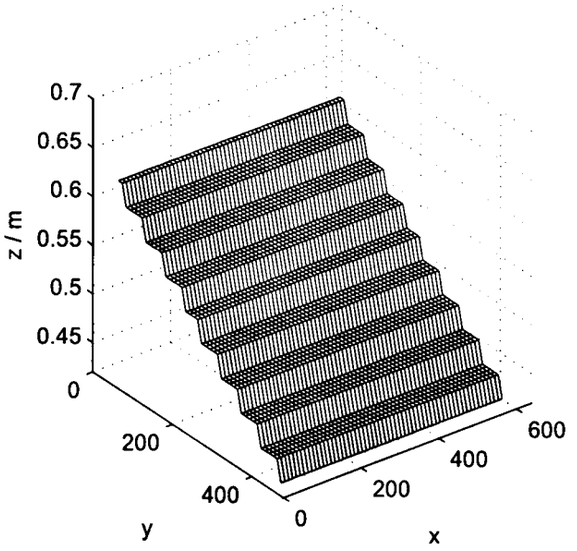


Figure 6.4: Actual depth map of the steps ranging from 0.42m to 0.62m

The resulting defocused images were corrupted with AWGN to give the required SNR and then they were converted to monochrome using an equal weighting of the colour planes. The noisy images were subsequently processed by the implementation of Ens and Lawrence's DFD algorithm incorporating the experimental PSF measurements. The MSE, mean and variance of the depth map were then calculated. For each SNR, five tests were performed with different realisations of the noise process and then the mean of the measures were recorded. The results of the analysis are presented in Table 6.1.

Table 6.1. Localisation results without and with (in brackets) median filtering

Window Size / pixels	SNR / dB	MSE / 10^{-4} m^2	Mean / 10^{-4} m	Variance / 10^{-4} m^2
32×32	40	4.49 (2.03)	4.77 (5.84)	4.49 (2.03)
	30	4.69 (2.13)	4.85 (5.38)	4.69 (2.12)
	20	6.14 (2.83)	4.85 (2.57)	6.14 (2.83)
64×64	40	0.996 (0.940)	-9.60 (-10.2)	0.987 (0.929)
	30	0.996 (0.939)	-9.68 (-10.3)	0.986 (0.929)
	20	1.02 (0.957)	-9.57 (-0.102)	1.01 (0.946)

The ratio of the MSEs for SNRs of 20dB to 40dB is 1.37 and 1.02 for window sizes of 32×32 and 64×64 respectively, thus showing that a larger window is less sensitive to noise. It is instructive to compare the depth maps shown in Figure 6.5. Although the MSE is lower for a 64×64 window compared to a 32×32 window (for all SNRs tested), the shape of the scene has clearly been lost.

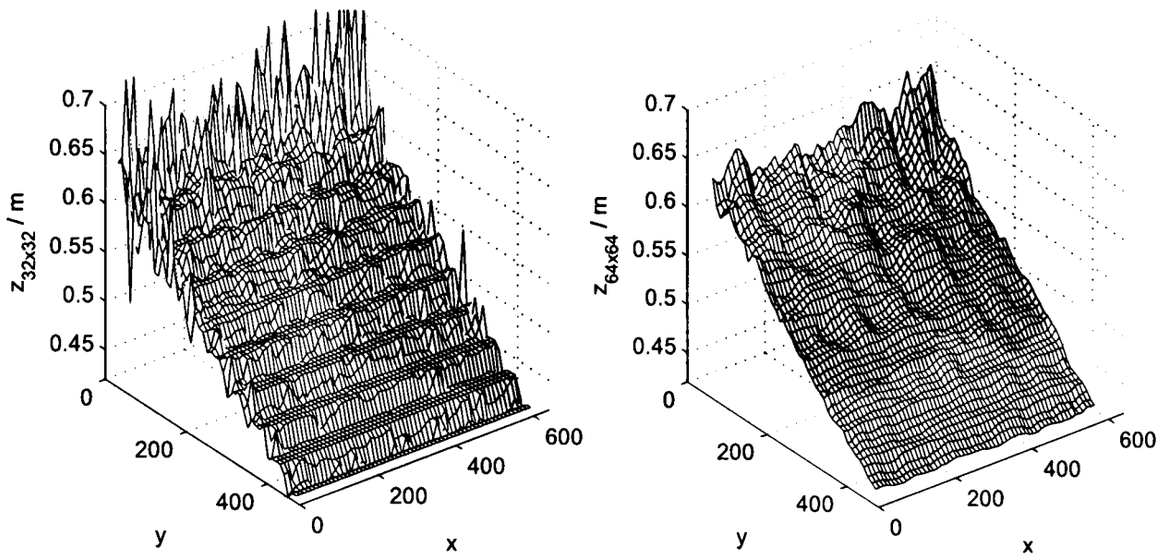


Figure 6.5: Depth maps produced by a 32×32 window (left) and a 64×64 window (right) for an SNR of 40dB

A step discontinuity is known to be the worst case depth profile and so a square wave depth map was set up with alternating strips of width $\delta = 48$ pixels and the actual depth map is shown in Figure 6.6. The MSE is a function of the depth of each side of the step, but for the purposes of simulation, depths of 0.42m and 0.50m were used. The result produced using a 32×32 window has a much lower MSE of 0.770 compared to 2.72 for a 64×64 window, shown in Figure 6.7.

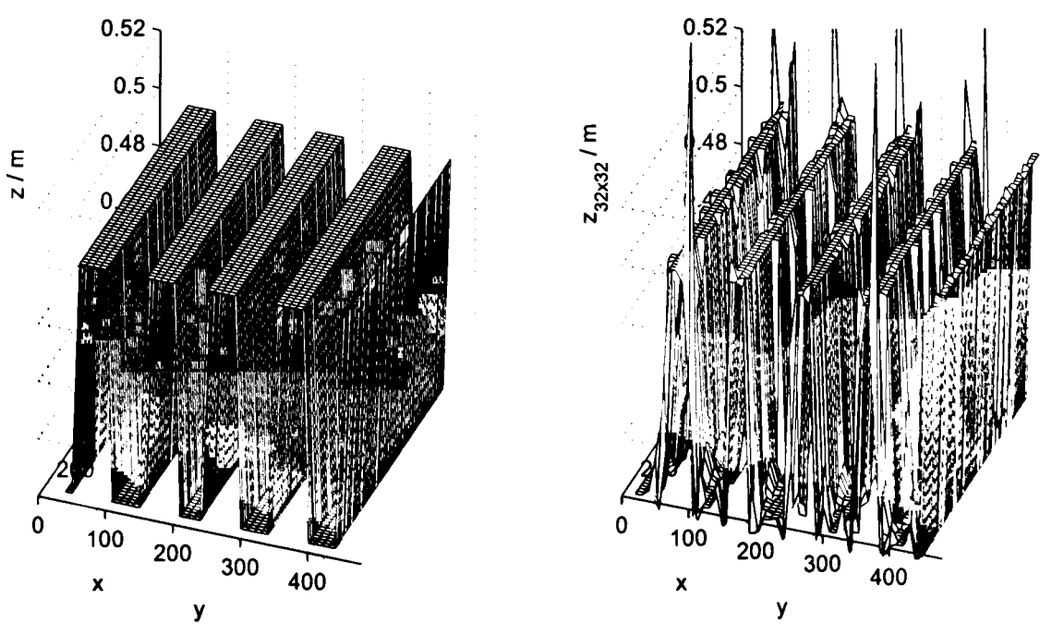


Figure 6.6: Actual depth map (left) and the result produced using a 32×32 window (right)

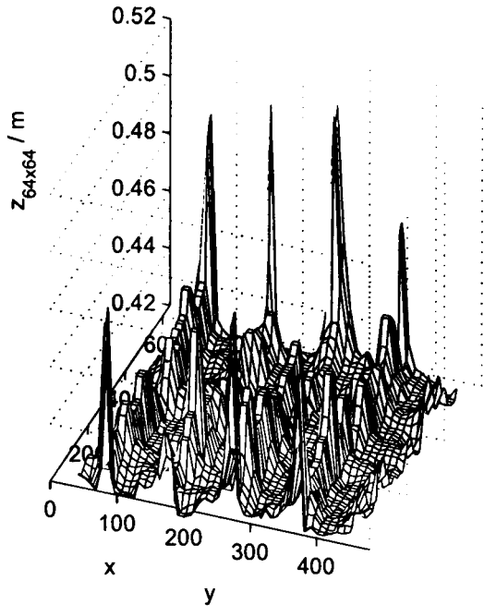


Figure 6.7: Depth map produced using a 64×64 window

In this experiment, no noise was added and the result using the 64×64 window was much worse than that produced using a 32×32 window. It was attributed to the fact that the 64×64 window was larger than the width of the constant depth region, thus the window straddles at least one, if not two, depth discontinuities.

In a further test, a square wave depth map was used where the depth alternated with strips of width δ pixels where $\delta = 2^n$ and $n = 0, 1, 2, ..., 8$. The depth maps were processed using a 32×32 window and Figures 6.8 and 6.9 show the MSE, mean and variance of the depth map as a function of δ. A total of 10,980 depth estimates using a pair of defocused images were calculated for a given δ to ensure a good estimate and only quantisation noise was present.

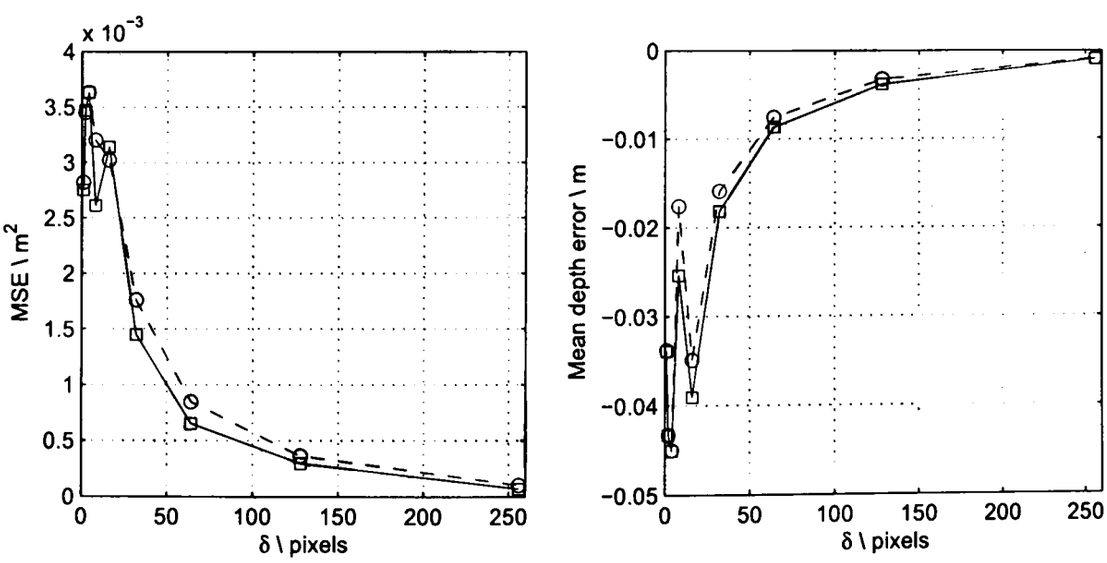


Figure 6.8: The MSE (left) and mean depth error (right) as a function of δ with (solid) and without (dashed) median filtering

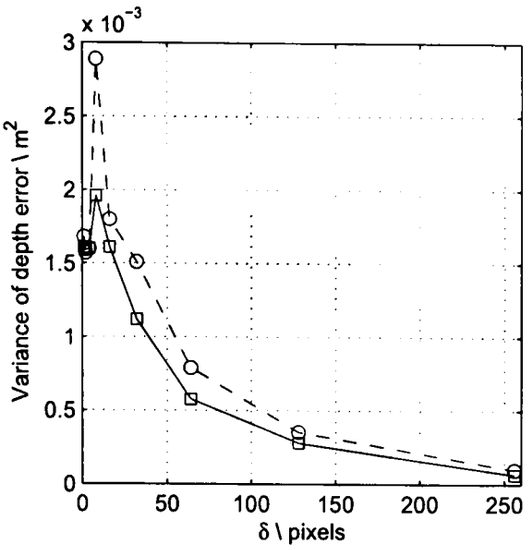


Figure 6.9: The variance of the depth error as a function of δ with (solid) and without (dashed) median filtering

As δ increases, there are less step discontinuities in the image and the image approximates a plane better. Ens and Lawrence's algorithm was based on the assumption that the depth in a window is constant and thus it was expected that the MSE would decrease with increasing δ as there are larger constant depth regions.

Generally the mean depth error approached zero as the scene became more like a plane with increasing δ. As discussed in Section 5.2.1, the depth is under-estimated if there is texture content due to closer objects in a window that is occluding the further depth object (which is required). As δ increases, this happens less often, thus reducing the mean error as expected.

Conclusion

A complete analysis of the trade-off between localisation and window size would need to take into account the depth discontinuities or depth profile, the texture of the surfaces, the camera parameters, the window size and the SNR. The analysis presented here has shown that if there are few depth discontinuities that a large window can be used to give robustness to noise. However, if there are lots of depth discontinuities then a small window could be used to recover the fine depth structure, but at the cost of poor SNR. Thus, the optimum window size will depend on the particular application that the DFD system is being used for, the likely range required and the type of objects in the scene.

6.2.12 Conclusion

The MATLAB implementation has been discussed along with the equation used to create simulated defocused images. A simple speed improvement has been shown. Averaging images captured by the camera reduces additive noise and for the conditions used about 8 images are required. Median filtering the depth maps produced by the DFD algorithm has been discussed and shown to decrease the noise in the output of the system.

The results show that the optimum aperture combination is either ($f_1 = 5.6$, $f_2 = 2.8$) or ($f_1 = 5.6$, $f_2 = 4$). The optimum error measure was found to be that used by Ens and Lawrence, namely the L_2-norm. To ensure good localisation and reasonable execution times the 32×32 image window was chosen and the fixed convolution ratio window of 21×21 pixels was found to perform better than a variable window. With the DFD algorithm set up the next sections examine each of the colour mixing algorithms.

6.3 Colour Mixing using a Genetic Algorithm with a Known Depth

6.3.1 Introduction

The research on colour DFD has examined whether there is an optimum combination of colour plane weightings (α, β, γ) such that the resulting monochrome image produces better depth estimates than simply using $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$. The research began with experiments using a GA with a known depth map to discover if there are optimum weights (α, β, γ) that could be evolved to reduce the depth map error produced by the DFD algorithm.

6.3.2 Practical Results

The colour checkerboard was pasted to a slope and imaged with the Basler A631fc colour camera using the 24mm Sigma photographic lens and apertures of f/5.6 and f/2.8. The slope had a depth that ranged from 0.440m to 0.520m and the statistics of the results are presented in Table 6.2. The depth maps after smoothing using a median filtering are presented in Figures 6.10 and 6.11.

Table 6.2. Results for GA with a known depth

Algorithm	Without Median Filtering			With Median Filtering		
	MSE / 10^{-3} m^2	Mean / 10^{-3}m	Variance / 10^{-3} m^2	MSE / 10^{-3}m^2	Mean / 10^{-3}m	Variance / 10^{-3}m^2
Mono	0.528	-3.98	0.512	0.208	-4.92	0.183
PCA	0.327	-0.168	0.327	0.112	-0.645	0.112
GA	0.0159	-0.634	0.0155	0.00372	-0.275	0.00364

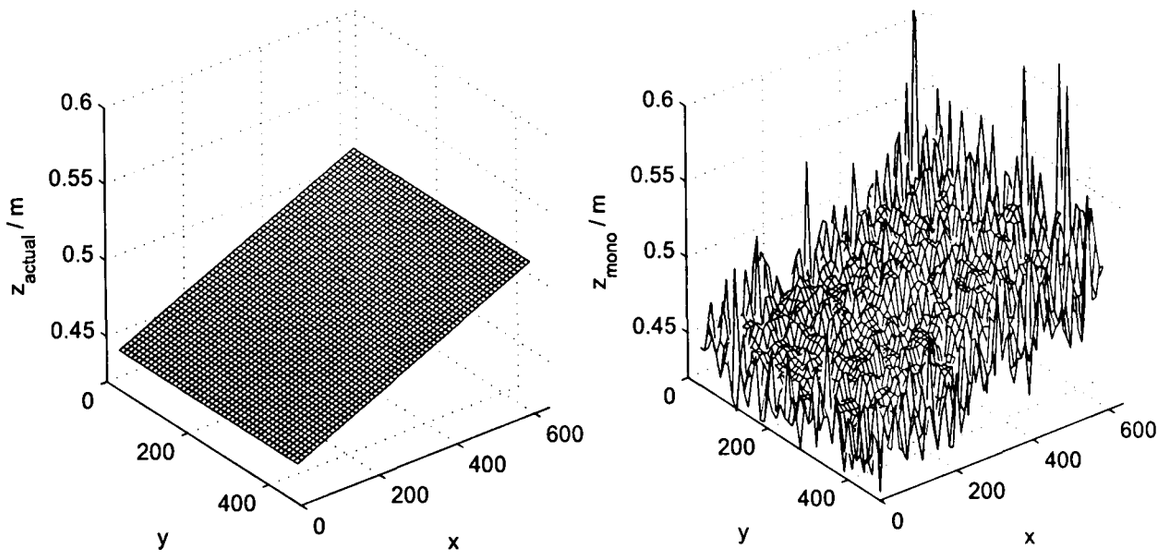


Figure 6.10: The actual depth map (left) and the result using the monochrome algorithm (right)

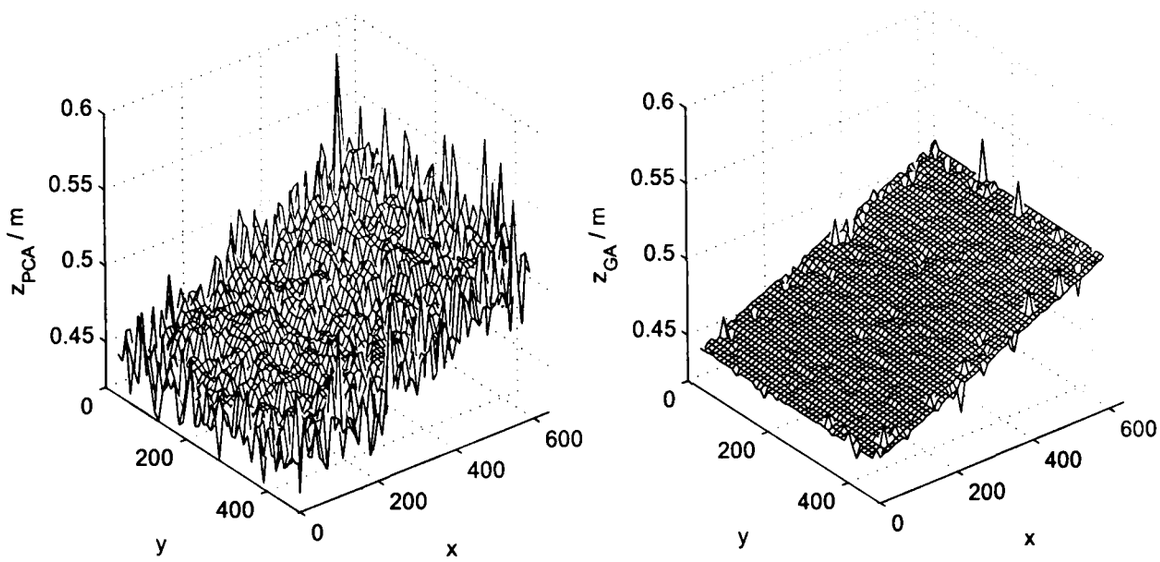


Figure 6.11: The depth map using PCA (left) and the GA with a known depth (right)

It can be readily seen that the GA has successfully managed to find the optimum scaling parameters (α, β, γ) to reduce the depth error and the MSE is 33 times lower than that using the monochrome case, i.e. where $\alpha = \beta = \gamma = \frac{1}{3}$, and 21 times lower than using PCA. The results of the monochrome and PCA algorithms have been given for compari-

son purposes and will be discussed in the following sections. In order to understand how the GA has produced such good results simulations were performed.

6.3.3 Simulated Results

An experiment was designed to illustrate the effect of the image noise and in particular how the GA can use the noise to improve the depth maps. A colour checkerboard pattern with randomly coloured 5×5 pixel squares was defocused using Equation (6.2) to simulate the texture being pasted to a plane at a depth of 0.520m. The GA requires the actual depth and in the first experiments the GA was given the depth of 0.520m. Each plane of the defocused images was independently corrupted with Additive White Gaussian Noise (AWGN) to give SNRs of 20, 30 and 40dB. The MSE of the recovered depth maps using the GA and the monochrome algorithm are given in Figure 6.12 and the mean depth errors are given in Figure 6.13. At a depth of 0.520m and an SNR of 20dB it is particularly noticeable that the MSE was better using the GA compared to the equal weighting case (mono). It could be argued that the GA has managed to reduce the noise level.

The experiment was re-run except that the depth given to the GA was incorrect. Depths of 0.470m, 0.495m, 0.545m and 0.570m were tested. It is very noticeable from the left hand side of Figure 6.12 that the increasing noise level has resulted in a reduction of the MSE for a given false depth. The mean depth error exhibited by the monochrome algorithm and shown in the right hand side of Figure 6.13 is what would be expected of the GA if it could not use the noise to give a lower mean error. The experiment showed that the GA is capable of using the noise present in the image to reduce the depth error without necessarily improving some property of the texture.

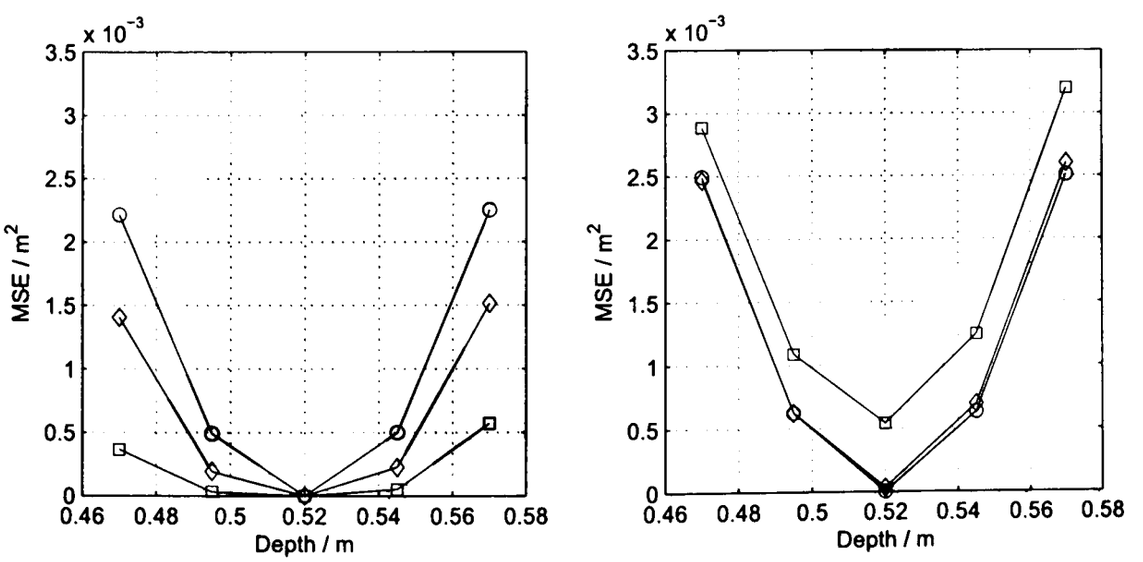


Figure 6.12: MSE results for the GA (left) and the monochrome case (right) with SNRs of 40dB (circle), 30dB (diamond) and 20dB (square)

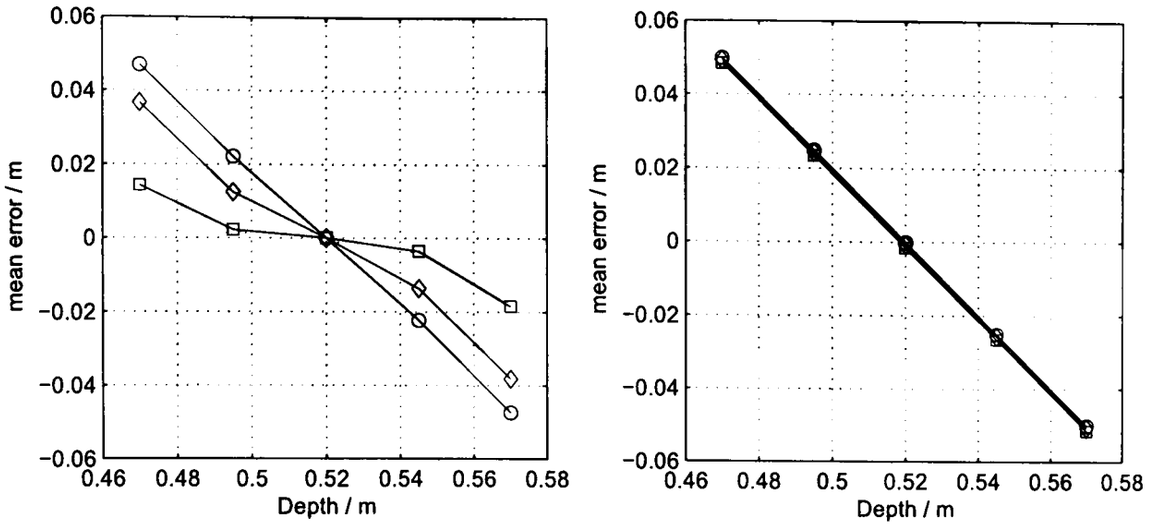


Figure 6.13: Mean error results for the GA (left) and the monochrome case (right) with SNRs of 40dB (circle), 30dB (diamond) and 20dB (square)

6.3.4 Conclusion

The results have shown that there are in fact optimum weights to reduce the depth error, but the workings of the GA are such that it cannot differentiate between signal and noise. The noise has been shown to help to produce depth maps with a lower MSE, even if the depth given to the GA is incorrect. The approach is not practical since if the depth was known then there would be no need to perform DFD. In the next four sections different algorithms are examined that produce colour plane weightings based on deterministic criteria that do not require the depth map to be known *a priori*.

6.4 Principal Component Analysis

6.4.1 Introduction

Principal Component Analysis (PCA) is a procedure for producing a 3×3 matrix of scaling constants to give three uncorrelated colour planes that are a linear combination of the original RGB colour planes. The planes are generally ordered in decreasing levels of variance and it is expected that the plane with the maximum variance is most useful for DFD work.

6.4.2 Simulation Results

A colour checkerboard with squares 5×5 pixels was defocus blurred to simulate a pattern pasted to a slope that had a depth that changes smoothly from 0.440m to 0.520m (left to right). The images that were used are shown in Figure 6.14.

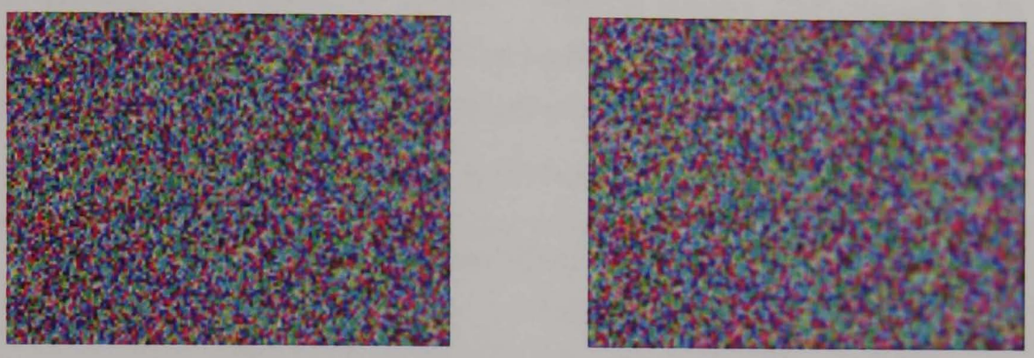


Figure 6.14: The simulated defocused slope images for f/5.6 (left) and f/2.8 (right)

The images were corrupted with AWGN and the three sets of scaling constants generated using PCA were used in turn to produce monochrome images that were then applied to DFD algorithm. The results of the experiment are presented in Table 6.3 for SNRs of 40dB (only quantisation noise present), 30dB and 20dB and where the first (P1), second (P2) and third (P3) principal planes were used.

Table 6.3. PCA results using a simulated, defocused colour checkerboard slope

	Mean Square Error / 10^{-3} m^2			
SNR / dB	Mono	PCA P1	PCA P2	PCA P3
40	0.00119	0.000937	0.00109	0.00153
30	0.0200	0.0137	0.0190	0.0280
20	0.250	0.162	0.229	0.358

The results show that at all noise levels the PCA algorithm using component 1 outperforms the monochrome case, and thus there appears to be some advantage to using a colour image for finding the accurate depth maps. As expected, the remaining two component planes (P2 and P3) formed using PCA have performed worse than the first component (P1), which can be attributed to their reduced variance. The first component has the largest variance of all three and the maximum signal-to-noise ratio [184].

At 40, 30 and 20dB the first principal component produces depth maps that are 1.3, 1.5 and 1.5 times better respectively than using the equal weighting algorithm, denoted the monochrome algorithm, where $(\alpha, \beta, \gamma) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$. Using the second principal compo-

nent reduces the improvement to 1.1 times for all SNR levels tested. The third principal component performed worse than the monochrome algorithm.

6.4.3 Experimental Results

In order to test the PCA algorithm practically a slope was set up that has a depth that changes smoothly from 0.440m to 0.520m. Different textures photographed using a 5 megapixel digital still camera were printed on a colour laser printer and pasted to the slope to provide a colour texture and the results are presented in Table 6.4.

Table 6.4. PCA results using the slope on five different textures

Texture	Algorithm	Without Median Filtering			With Median Filtering		
		MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Carpet (carpet_01)	Mono	0.202	-1.73	0.199	0.0755	-2.37	0.0699
	PCA	0.239	1.89	0.236	0.0832	1.19	0.0818
Colour C.B.	Mono	0.456	-3.17	0.446	0.210	-3.88	0.195
	PCA	0.320	1.16	0.318	0.141	0.623	0.140
Grass (grass_02)	Mono	0.261	-0.668	0.261	0.0958	-1.37	0.0939
	PCA	0.286	1.90	0.282	0.104	1.22	0.102
Red stone (stone_03)	Mono	1.02	0.117	1.02	0.295	-2.20	0.290
	PCA	1.08	3.31	1.07	0.322	0.778	0.321
Stone (stone_08)	Mono	0.253	-8.38	0.183	0.150	-8.88	0.0715
	PCA	0.230	-6.18	0.191	0.122	-6.74	0.0768

Of the five tests, only the colour checkerboard (CB) and the red stone texture enabled the PCA method to produce depth maps with a lower MSE than the monochrome case. For the cases of the checkerboard and red stone, PCA produced MSEs that were 1.4 and 1.1 times lower than using monochrome. For the remaining textures, the monochrome algorithm outperformed PCA by between 1.1 and 1.2 times. Thus, the practical results are in direct conflict with the simulation results where the PCA algorithm performed better than the monochrome case, even in fairly high noise levels. The monochrome algorithm generally under-estimated the depth and as the PCA over-estimates the depth it appears that it is boosting the noise (as discussed in Section 5.2.2).

Cameras are essentially photon counting devices and the underlying distribution is usually assumed to be the Poisson model. Consider a single photosite on the CCD where

a photons are collected. Then the probability that k photos are counted at the photosite is given by

$$P(a = k) = \frac{e^{-\lambda}\lambda^k}{k!} \tag{6.12}$$

where $k = 0, 1, 2, \ldots$ [185]. The variance of the Poisson distribution is equal to its expected value or mean and so as the brightness increases, so does the noise level.

Corner *et al.* [184] analysed the effect of additive and multiplicative noise on Landsat images that were processed using PCA. They showed experimentally that PCA can separate additive, normally distributed, uncorrelated noise from the signal and the ability is degraded if the noise is correlated between the channels. Rosipal *et al.* [186] explained that adding white noise is equivalent to adding a diagonal matrix to the covariance matrix, with the noise variances of each channel appearing along the diagonal. For isotropic noise this will lead to the same increase of all eigenvalues and if the SNR is sufficiently high then only the principal components corresponding to the smaller eigenvalues will be strongly affected.

Corner *et al.* [184] modelled the multiplicative noise as a unit mean, normally distributed random process with a probability density function (PDF) of

$$f(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left\{-\frac{1}{2}\frac{(x-1)^2}{\sigma^2}\right\} \tag{6.13}$$

and if the noise-free signal is denoted $s(x, y)$ then the degraded, noisy signal $d(x, y)$ is given by

$$d(x, y) = s(x, y)\, n(x, y) \tag{6.14}$$

where $n(x, y)$ is multiplicative noise. They found that for $\sigma = 1.0$ nearly all of the signal and noise were contained in the first component. For $\sigma > 1.0$ the performance of PCA decreases rapidly.

Green *et al.* [187] found that PCA does not always produce images with decreasing image quality with increasing component number. One solution suggested was to rescale the data so that the bands have equal noise variance and then perform PCA, but this requires the noise variances to be known. This approach will be known as Noise Variance Adjusted PCA (NVA-PCA). If the noise variances are equal then PCA can be performed without any scaling.

One thousand images were taken of the colour checkerboard pattern and then for a given pixel (x, y) in all of the images the mean and variance of the brightness of each colour plane was calculated. The results for all of the pixel positions were collected and the results are illustrated in Figure 6.15. Experiments performed on the camera suggested that multiplicative noise was dominant over the additive noise, as the variance is a func-

tion of brightness. Withagen *et al.* [188] showed in their experiments that multiplicative noise exceeded additive noise at around 10 to 30% of the intensity range for their cameras using a similar technique. Further, from the results here $\sigma > 1$ and thus by the work of Corner *et al.* [184] it may be concluded that the poor results are due to PCA's inability to work well in high levels of multiplicative noise.

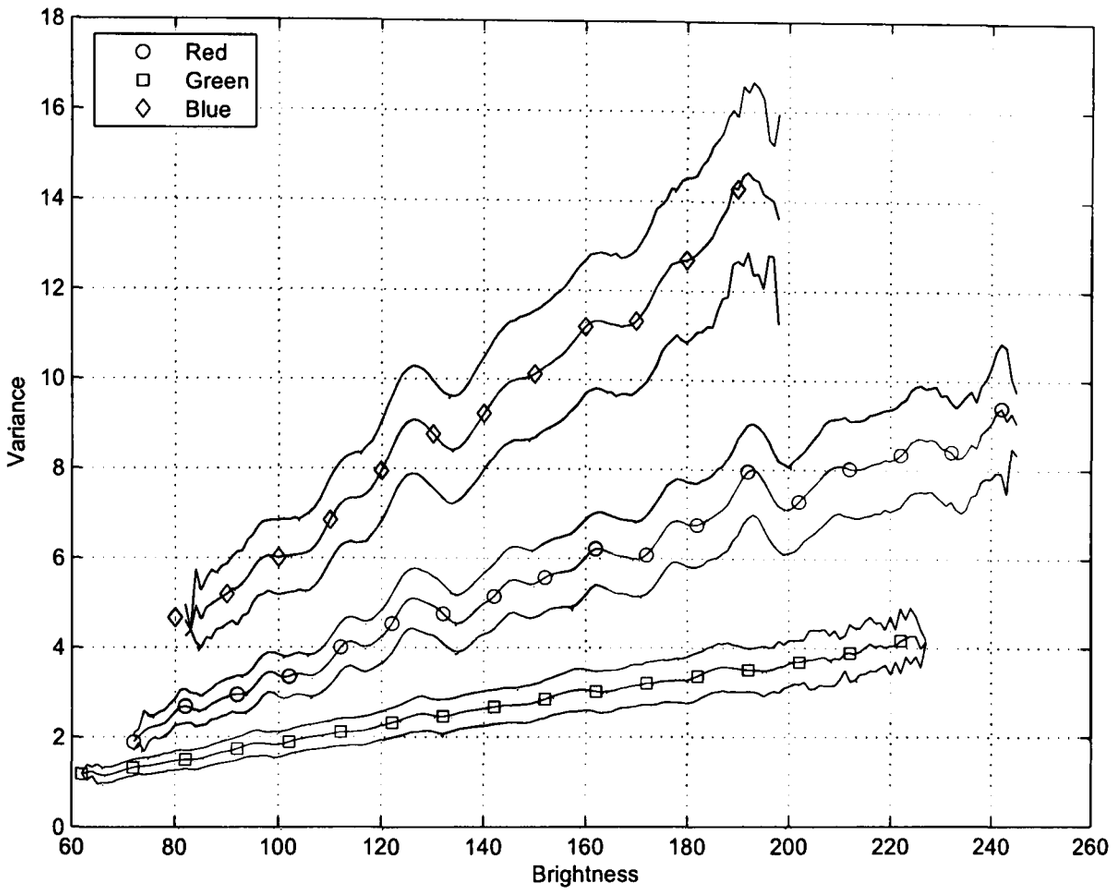


Figure 6.15: The noise variance and mean as a function of brightness (mean and 3 standard deviations shown)

The relative variances of the colour planes were determined by fitting a straight line to the data presented in Figure 6.15 and the relative variances are shown in Table 6.5.

Table 6.5. Relative variances of the noise for each colour plane

Channel	Relative Variance
Red	0.54
Green	0.28
Blue	1

It can be seen from Table 6.4 that generally the depth map produced using the equal weighting algorithm (Mono) under-estimates the depth, whereas PCA over-estimates the depth. The scaling parameters (α, β, γ) were calculated from the more focused image (i.e. that taken with f/5.6). As discussed above, PCA emphasises multiplicative noise and so it appears that the noise in image 1 was been boosted relative the second image, thus over-

estimating the depth (as discussed in Section 5.2.2). The depth map returned using the monochrome algorithm has been under-estimated, which is consistent with image 2 having a lower SNR than image 1 and this is caused by the reduction of the variance of the texture due to increased defocusing, compared to image 1.

The same images used to create the depth statistics in Table 6.4 were used and the results using the monochrome, PCA and NVA-PCA algorithms are presented in Table 6.6. The results of using only one of the colour planes, i.e. red (R), green (G) or blue (B), are also presented for comparison as it clearly highlights the effects of the noise.

Table 6.6. Results using the slope with pasted textures

	Mean Square Error (without median filtering) / $10^{-3}m^2$					
Texture	Mono	PCA	NVA-PCA	R	G	B
Carpet	0.202	0.239	0.395	0.612	0.624	1.36
Colour C.B.	0.456	0.320	0.282	0.253	0.321	1.90
Grass	0.261	0.286	0.357	0.479	0.464	1.02
Red stone	1.02	1.08	1.14	1.32	1.25	1.91
Stone	0.253	0.230	0.259	0.396	0.390	1.61

The results of the monochrome and PCA algorithms are the same as in Table 6.4, but they have been reproduced here for ease of comparison. The NVA-PCA algorithm produced better results than PCA on the colour checkerboard only, which was disappointing.

Processing the blue colour plane only has resulted in MSEs that are always worse than using either the red or green planes. This can be attributed to the much higher noise level and the texture in the blue plane had the lowest variance, as shown in Table 6.7.

Table 6.7. Variances of the RGB colour planes of the more focused image

Image	Var[R] / 10^{-3}	Var[G] / 10^{-3}	Var[B] / 10^{-3}
Carpet	4.64	3.81	2.14
Colour checkerboard	8.81	6.05	3.19
Grass	5.93	8.94	3.05
Red stone	10.6	9.37	4.52
Stone	5.66	6.27	4.07

The plane with the highest variance correlates with the lowest MSE in all of the textures tested, except for the red stone where the variance of the red plane is greater than that of the green plane, but the MSE using the green plane is lower. The effect can be attributed to the fact that the variance of the green plane is only slightly lower than that of the red, but the noise variance is lowest for the green. Thus, the maximum SNR occurs

with the green plane. Yuan and Subbarao [83] suggested using the plane with the maximum variance and the results suggest this would not be optimum. A simpler and more accurate solution is to use an equal weighting of the colour planes, as shown by comparing the single colour plane results with the monochrome results in Table 6.6.

6.4.4 Conclusion

The first principal component produced using PCA was found to be better than the remaining two and the equal weighting algorithm in simulations on a colour checkerboard pattern with AWGN. PCA generally produced worse results than the monochrome algorithm in practical experiments and this was found to be due to the strong multiplicative noise component. The colour planes were weighted based on their noise variance in an algorithm denoted NVA-PCA, but this adjustment did not help. The MSE using a single colour plane (i.e. either red, green or blue) tallies well with the plane with the maximum variance, as would be expected as it has the maximum SNR. Overall the results suggest that a new formulation of PCA is required that accounts for the multiplicative noise that occurs in practice.

6.5 SNR Maximisation Algorithm

6.5.1 Introduction

The signal-to-noise ratio (SNR) maximisation algorithm assumes an additive noise model and requires the variance of the image segment being processed as well as that due to the noise. The statistics of the noise are not simple to find and approximations have to be made. The trivial solution is to assume that there is no noise present and thus any set of (α, β, γ) is optimum as long as the resulting image possesses some texture. The first approximation is to assume quantisation noise only exists, which is additive and has the same variance in each plane.

The SNR maximisation algorithm was designed before the actual camera (the Basler A631fc colour camera) was obtained, and as it has a strong multiplicative noise component only simulations could be performed to evaluate the theory. Unfortunately, the noise model of the algorithm was incorrect for the hardware. If the actual noise had been additive then experimental results could have been reported.

6.5.2 Simulated Experimental Results

The formulation of the measure of the SNR assumes an additive model and so in the initial experiments noise with a constant variance was added to a simulated defocused image. It was assumed that the noise process occurs after the lens. Each image has an associated SNR and as defocus decreases the variance of the texture then it follows that the most defocused image has the lowest SNR. Tests were performed where the SNR used as the objective value in the GA was either the SNR due to the first image or the second image or the mean SNR of both images. The colour checkerboard with randomly coloured 5×5 pixel squares was used as a texture for the example and the SNR set to 40dB, 30dB and 20dB. The results are presented in Tables 6.8 to 6.10 where the mean SNR of the monochrome image resulting from colour mixing is given along with the depth performance parameters.

Table 6.8. Checkerboard results with an SNR of 40dB (same noise variances in each colour plane)

	Mean SNR / dB		Without Median Filtering		
Algorithm	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	44.20	40.84	0.00119	0.103	0.00118
PCA	45.39	42.31	0.000937	0.0903	0.000929
Max SNR (1)	45.15	42.05	0.000979	0.0739	0.000974
Max SNR (2)	45.05	42.20	0.00107	0.0914	0.00106
Max SNR (Ave)	45.13	42.14	0.000937	0.0903	0.000929

Table 6.9. Checkerboard results with an SNR of 30dB (same noise variances in each colour plane)

	Mean SNR / dB		Without Median Filtering		
Algorithm	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	29.60	26.25	0.0200	-0.331	0.0199
PCA	30.80	27.71	0.0137	-0.257	0.0136
Max SNR (1)	30.80	27.71	0.0137	-0.258	0.0136
Max SNR (2)	30.67	27.89	0.0160	-0.220	0.0159
Max SNR (Ave)	30.78	27.83	0.0148	-0.230	0.0147

Table 6.10. Checkerboard results with an SNR of 20dB (same noise variances in each colour plane)

Algorithm	Mean SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	19.60	16.25	0.250	-2.67	0.243
PCA	20.80	17.71	0.162	-1.78	0.159
Max SNR (1)	20.80	17.71	0.162	-1.77	0.159
Max SNR (2)	20.67	17.89	0.162	-1.75	0.186
Max SNR (Ave)	20.78	17.83	0.166	-1.55	0.164

Although only one set of results are presented for each SNR, the results are indicative of the efficacy of the algorithms and this was because each depth map was composed of a large number of points (in this case 2,745 individual depth measurements).

When the noise variance in each colour plane is the same there is no improvement in maximising the SNR over using the first principal component created using PCA. Except for the case of only quantisation noise present (40dB), maximisation of the SNR using image 1 performed better than the other two alternatives, namely using 2 or the mean SNR.

The noise variances of each colour plane are not identical in practice. Due to averaging of the green pixels on the Bayer filter the green plane has the lowest variance and the efficiency of the semiconductor to blue light means that the blue must be amplified the most, leading to the largest variance. From a practical experiment it was found that the relative variance of the red plane was about 0.54 of that due to the blue and 0.28 for the green plane. In the simulation experiments reported in Tables 6.11 to 6.13 AWGN corrupted each image and the variance was different in each plane and dictated by the practically obtained ratios.

Table 6.11. Checkerboard results with an SNR of 30dB (different noise variances in each colour plane)

Algorithm	Mean SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	31.78	28.43	0.0361	-0.640	0.0357
PCA	33.18	30.10	0.00967	-0.0962	0.00966
Max SNR (1)	35.23	31.96	0.00562	-0.00477	0.00562
Max SNR (2)	35.14	32.06	0.00703	-0.0423	0.00703
Max SNR (Ave)	35.22	32.01	0.00656	0.0342	0.00656

Table 6.12. Checkerboard results with an SNR of 20dB (different noise variances in each colour plane)

Algorithm	Mean SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	21.83	18.54	0.486	-4.86	0.462
PCA	23.16	20.08	0.110	-1.20	0.109
Max SNR (1)	25.23	21.97	0.0627	-0.423	0.0625
Max SNR (2)	25.14	22.06	0.0751	-0.665	0.0746
Max SNR (Ave)	25.22	22.02	0.0657	-0.368	0.0656

Table 6.13. Checkerboard results with an SNR of 10dB (different noise variances in each colour plane)

Algorithm	Mean SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	12.29	9.499	3.65	-27.7	2.88
PCA	12.92	9.976	2.04	-15.0	1.81
Max SNR (1)	15.25	12.03	1.06	-5.38	1.03
Max SNR (2)	15.14	12.14	1.47	-7.97	1.40
Max SNR (Ave)	15.23	12.09	1.16	-5.95	1.13

At SNRs of 30dB, 20dB and 10dB the maximisation of the SNR has improved the SNR by 3.5dB, 3.4dB and 3.0dB respectively compared to the monochrome case. PCA has an inherent ability to improve the SNR, but the algorithm incorporating the GA has managed to improve the SNR by between 2.1dB and 2.3dB compared to using PCA. The mean error has been under-estimated in all cases except one, which was expected due to the analysis presented in Section 5.2.2.

When the noise variance is not identical in each plane, PCA (using the component with the largest eigenvalue) has consistently produced better MSEs than the monochrome algorithm. In the tests, the maximisation of the SNR using a GA to evolve the solution has produced better results than either the monochrome or PCA. Improvements of 6.4, 7.8 and 3.4 times were found using SNRs of 30dB, 20dB and 10dB respectively compared to using the equal weighting (mono) algorithm. Maximising the SNR produced improvements of 1.7, 1.8 and 1.9 times compared to using PCA with SNRs of 30dB, 20dB and 10dB respectively. Thus, the results have shown that with increasing noise variance (i.e. decreasing SNR), maximisation of the SNR produces increasingly better results compared to PCA.

Figures 6.16 and 6.17 show the depths map produced when the SNR was 20dB and the noise variance of each plane was different.

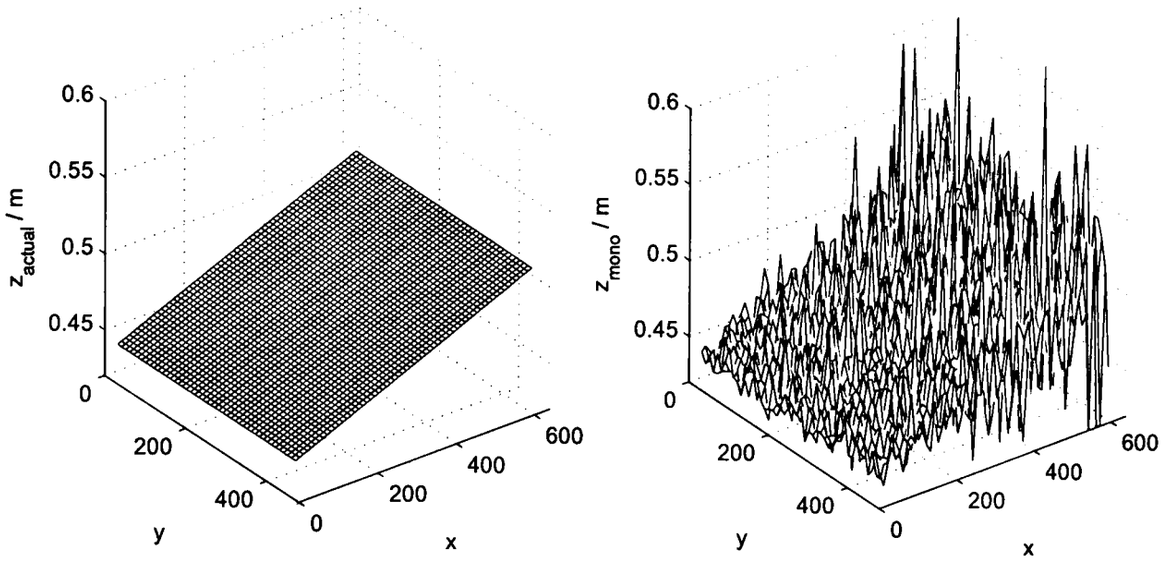


Figure 6.16: Actual depth map (left) and that produced using equal weighting (right) when the SNR was 20dB

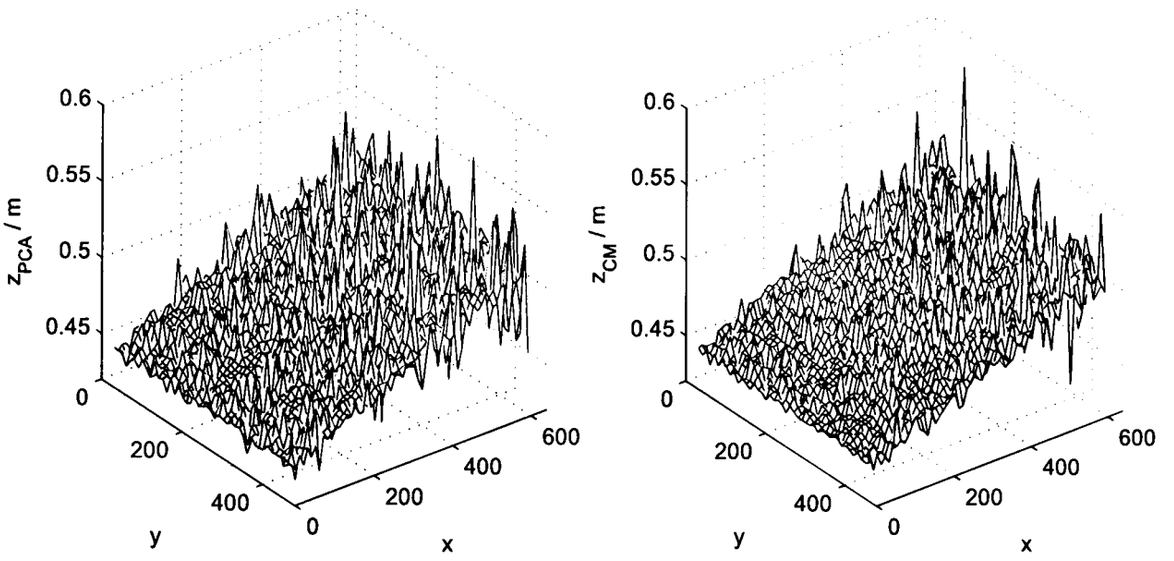


Figure 6.17: Depth maps produced using PCA (left) and maximisation of the SNR (right)

A particularly notable feature of the depth maps is that the errors increase as the depth increases and this can be attributed to the fact that the SNR decreases with distance due to the decreased signal variance caused by increased defocusing.

6.5.3 Conclusion

The simulation results showed that maximising the SNR can lead to reductions in the error of the depth map and especially so if the noise variance is not the same in each colour plane (i.e. it is non-isotropic), as expected in practice. Improvements of between 3.4 and 7.8 times were found using the algorithm compared to the $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$ case and between 1.7 and 1.9 times compared to the first principal component produced using PCA.

The real camera system used had a strong multiplicative noise component, which meant that the algorithm could not be tested on real data. The SNR maximisation algorithm requires the variance of the noise of each colour plane to be known and it was shown in Section 6.4.3 that the variance is a function of brightness. Therefore, no single variance value exists for a given colour plane, thus making the approach unusable in practice.

6.6 Fractal Dimension Maximisation

6.6.1 Introduction

The fractal dimension (FD) of a surface is a measure of its roughness and thus for an image it is a measure of the brightness variation and hence texture. It is known that a textureless surface is useless for DFD work and so if the roughness of the surface can be increased then maybe the depth estimate should be better. In the next section simulation results are presented.

6.6.2 Simulated Experimental Results

In the first two experiments presented here, shown in Table 6.14 and Table 6.15, the colour checkerboard and grass texture were defocused in software to simulate being at 0.50m. No noise was added so that the only noise was quantisation noise, thus giving an SNR of around 40dB.

Table 6.14. Colour checkerboard simulated to be at 0.50m

Algorithm	SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	42.20	38.81	0.00645	0.318	0.00635
PCA	43.62	40.61	0.00510	0.578	0.00477
Max FD	41.96	38.33	0.00739	-0.0156	0.00739

Table 6.15. Grass texture simulated to be at 0.50m

Algorithm	SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	40.64	37.85	0.0136	0.901	0.0128
PCA	40.90	38.16	0.0140	0.953	0.0131
Max FD	35.57	32.61	0.0312	0.760	0.0306

The results show that the maximisation of the fractal dimension (Max FD) algorithm produced worse MSEs and the reason can be traced to the reduction in the SNR for images 1 and 2 through the colour mixing. Just over 5dBs were lost by maximising the FD compared to using the monochrome or PCA algorithms. To further show the effect of noise the carpet texture was defocused to simulate being placed 0.50m from the camera and the resulting images were corrupted with AWGN to give an SNR of 30dB and the results are shown in Table 6.16.

Table 6.16. Carpet texture simulated to be a 0.50m with a nominal SNR of 30dB

Algorithm	SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	32.25	29.09	0.0746	0.500	0.0744
PCA	32.33	29.22	0.0817	0.526	0.0814
Max FD	27.61	24.32	0.284	-0.474	0.283

In the results of Table 6.16 the noise variance was set to be the same for each plane, but the results showed the same pattern even when noise of differing variances was used. An SNR of 30dB would be considered good for an image processing system and yet maximising the FD has clearly resulted in a poor noise performance as the SNR has been reduced compared to the other algorithms. Clearly the maximisation of the fractal dimension

algorithm is not suitable for use in colour mixing for DFD due to the inherent noise reduction as shown by the results in Table 6.17 for SNRs of 40, 30 and 20dB.

Table 6.17. Random checkerboard pattern pasted on a slope

Algorithm	SNR / dB		Without Median Filtering		
	Image 1	Image 2	MSE / $10^{-3}m^2$	Mean / $10^{-3}m$	Variance / $10^{-3}m^2$
Mono	44.20	40.84	0.00119	0.103	0.00118
PCA	45.39	42.31	0.000937	0.0903	0.000929
Max FD	44.02	40.59	0.00117	0.0688	0.00117
Mono	29.60	26.25	0.0200	-0.331	0.0199
PCA	30.80	27.71	0.0137	-0.257	0.0136
Max FD	29.27	25.81	0.0210	-0.497	0.0207
Mono	19.60	16.25	0.250	-2.67	0.243
PCA	20.80	17.71	0.162	-1.78	0.159
Max FD	19.18	15.72	0.294	-3.68	0.280

6.6.3 Conclusion

The fractal dimension of a surface gives a measure of its roughness and changing (α, β, γ) has resulted in worse depth estimates, even with a very low noise level, compared to the monochrome case or using PCA. It was later found that the formulation of measuring the FD using a least squares fit to the fBm model was too sensitive to noise [172] [173] [174] and consequently it was not suitable for DFD.

6.7 Localisation through Colour Mixing

6.7.1 Introduction

The *Localisation through Colour Mixing* (LCM) algorithm seeks to find the optimum (α, β, γ) to extract the blurring contribution due to the central pixel only. The experiments could only be performed in simulation due to the lack of availability of a projector with a telecentric aperture. The specially modified projector is to ensure that the size of the squares, as seen by the camera, do not change with distance.

6.7.2 Simulation Experiments

A Genetic Algorithm was used to evolve the optimum colour pattern and it was tiled to create a 640×480 image. Each pixel in a 32×32 window had a different colour. The pinhole image created was then defocus blurred to simulate being placed on 10 steps that equally spanned the range 0.42m to 0.62m. In the first experiment presented in Table 6.18 the LCM algorithm was tested when the less defocused image was used to find (α, β, γ), denoted LCM (1), the most defocused image, denoted LCM (2), and the pinhole image, denoted LCM (P). The pinhole image was in fact the projected image.

Table 6.18. Ten steps in the range [0.42, 0.62] with only quantisation noise present

Algorithm	Without Median Filtering			With Median Filtering		
	MSE $/ 10^{-3}m^2$	Mean $/ 10^{-3}m$	Variance $/ 10^{-3}m^2$	MSE $/ 10^{-3}m^2$	Mean $/ 10^{-3}m$	Variance $/ 10^{-3}m^2$
Mono	2.45	-5.39	2.42	0.980	-3.23	0.969
PCA	0.582	4.49	0.561	0.254	2.56	0.247
LCM (1)	0.335	-0.356	0.334	0.145	-0.684	0.144
LCM (2)	0.261	-2.16	0.256	0.131	-2.05	0.126
LCM (P)	0.861	0.925	0.861	0.278	0.360	0.278

The results show that the LCM using image 2 possesses a lower MSE than that of the monochrome and PCA algorithms with and without median filtering. LCM using images 1 and 2 were 7.3 and 9.4 times better than Mono and 1.7 and 2.2 times better than PCA respectively. It is interesting to examine the SNRs, which are shown in Table 6.19. It is clear that maximising the localisation has resulted in a drop in the average SNR compared to PCA and monochrome and thus there appears to be a trade-off in action. The SNRs using the LCM algorithms are all less than both the monochrome and PCA cases, but the MSEs of the depth maps are lower.

Table 6.19. SNRs following colour mixing

Algorithm	Image 1 / dB	Image 2 / dB
Mono	31.32	28.16
PCA	33.59	29.78
LCM (1)	31.87	26.69
LCM (2)	31.68	26.26
LCM (P)	32.08	27.26

One of the problems with the evolved texture is that the SNR is only about 30dB when quantisation noise is present, which is quite low. Experiments were performed where the

pinhole image was set to be randomly coloured squares that were set to different sizes. It was found that squares of 5×5 pixels worked well and thus a randomly coloured pattern was used for the remainder of the LCM experiments. A GA was not required to evolve the pattern because the optimum texture was found to be a random pattern.

To illustrate the effect of the trade-off in the next experiment presented, 10 steps were simulated that equally span the depth range [0.42, 0.62]. The noise-free images are shown in Figure 6.18 and the results are given in Table 6.20.

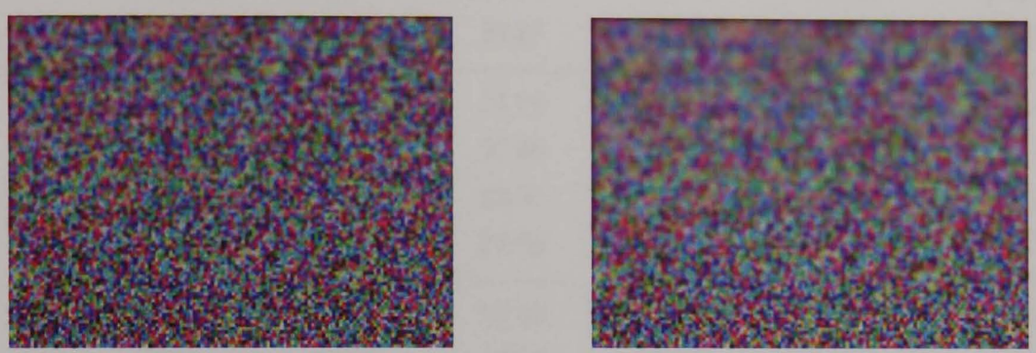


Figure 6.18: The simulated defocused steps for f/5.6 (left) and f/2.8 (right)

By using larger squares, the SNR with only quantisation noise increased from 30dB to 40dB compared to using the random coloured pattern with 1×1 pixel squares. With only quantisation noise present and 5×5 pixel squares the LCM algorithm using image 1 has a MSE that is 4.8 times lower than monochrome and 2.7 times lower than using PCA. However, with an SNR of 30dB the improvement has dropped to just 1.1 times lower and at 20dB LCM is performing the worst. PCA's ability to improve the SNR in these experiments is clearly giving it an advantage that outweighs that due to localisation.

Table 6.20. Step results for depth range [0.42, 0.62]

Nominal SNR / dB	Algorithm	SNR / dB Image 1	SNR / dB Image 2	Without Median Filtering MSE / $10^{-3}m^2$	Without Median Filtering Mean / $10^{-3}m$	Without Median Filtering Variance / $10^{-3}m^2$
40	Mono	41.73	38.16	0.552	0.877	0.552
	PCA	43.08	39.87	0.310	0.905	0.309
	LCM (1)	41.36	37.51	0.114	-0.158	0.114
	LCM (2)	41.29	37.27	0.0993	-0.480	0.0991
30	Mono	28.91	25.35	0.734	0.334	0.734
	PCA	30.27	27.06	0.454	0.407	0.454
	LCM (1)	28.55	24.70	0.404	-0.873	0.403
	LCM (2)	28.47	24.46	0.360	-1.83	0.357
20	Mono	18.91	15.35	2.24	-3.34	2.23
	PCA	20.27	17.06	1.63	-1.23	1.62
	LCM (1)	18.54	14.70	2.45	-6.60	2.41
	LCM (2)	18.47	14.48	2.46	-9.07	2.37

Using the 5×5 pixel pattern LCM (2) has produced MSEs that are 5.6 and 2.0 times better than Mono and 3.1 and 1.3 times better than PCA at SNRs of 40 and 30dB respectively. At 20dB all algorithms performed badly and PCA produced the best results.

The interesting aspect of the results is that although the 5×5 pixel squares ensured the image has a better SNR compared to using the evolved pattern where every square has a different colour, LCM produced much better results compared to PCA and Mono. At 40dB the improvement was 9.4 times better than monochrome using the evolved pattern, whereas the 5×5 pattern decreased the improvement to 5.6 times. Thus, clearly the 1×1 pixel squares is better for LCM, but worse for Mono and PCA.

In Figure 6.19 and Figure 6.20 the depth map results without median filtering are presented for the 40dB case.

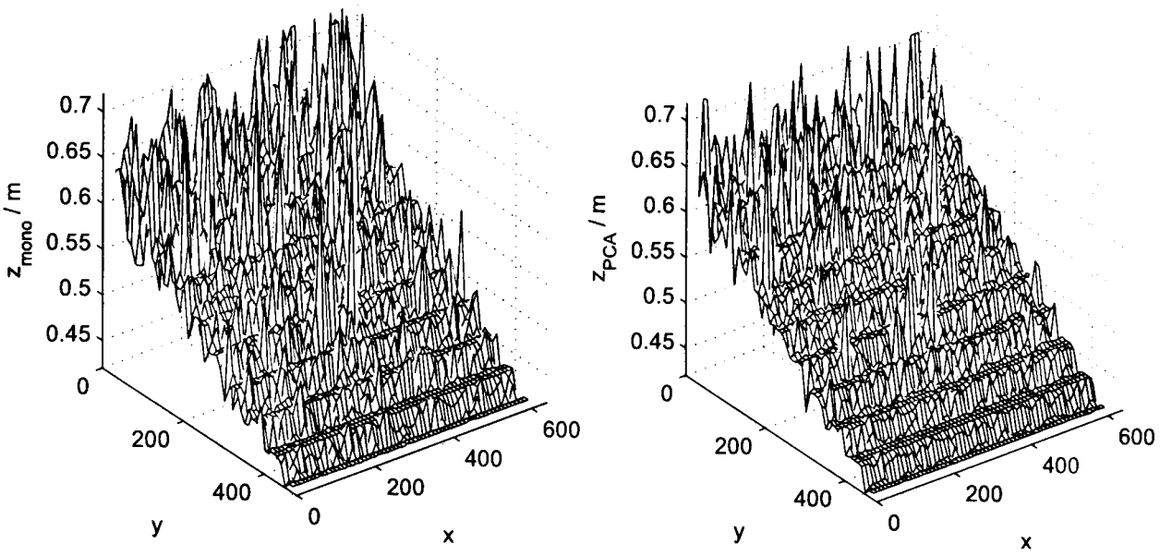


Figure 6.19: Monochrome and PCA results

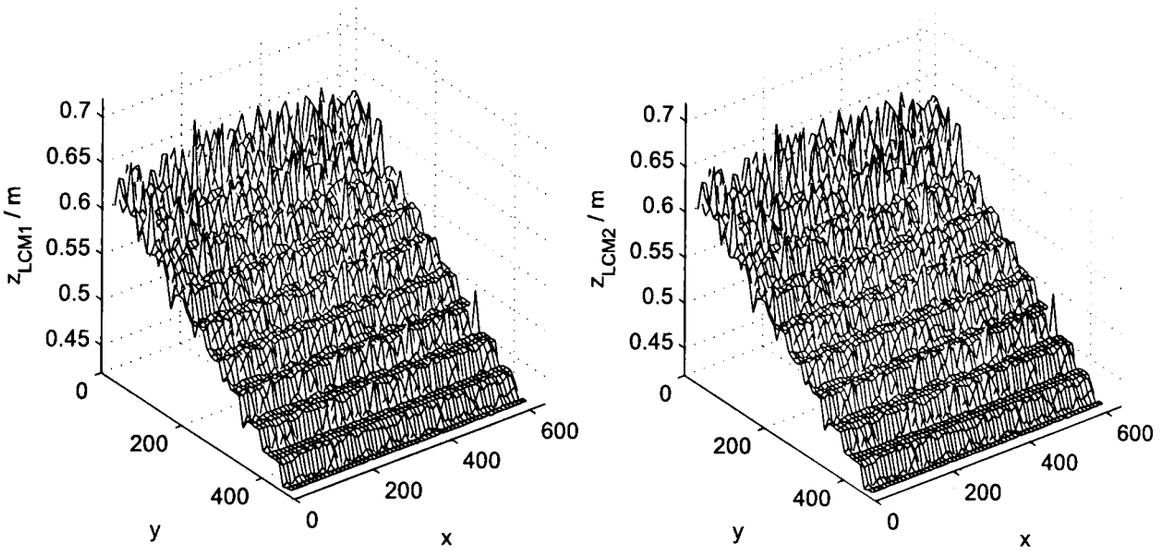


Figure 6.20: Depth maps using LCM (1) and LCM (2) algorithms

LCM has produced lower error at larger depths, which can be attributed to the reduction in the image overlap problem, which was discussed in Section 2.4.1. The images shown in Figure 6.21 show a particular 32×32 segment that was processed using equal weighting, PCA and LCM. It is very noticeable that the LCM algorithm has localised the blurring effect of the central pixel very well compared to that produced using the monochrome and PCA algorithms.

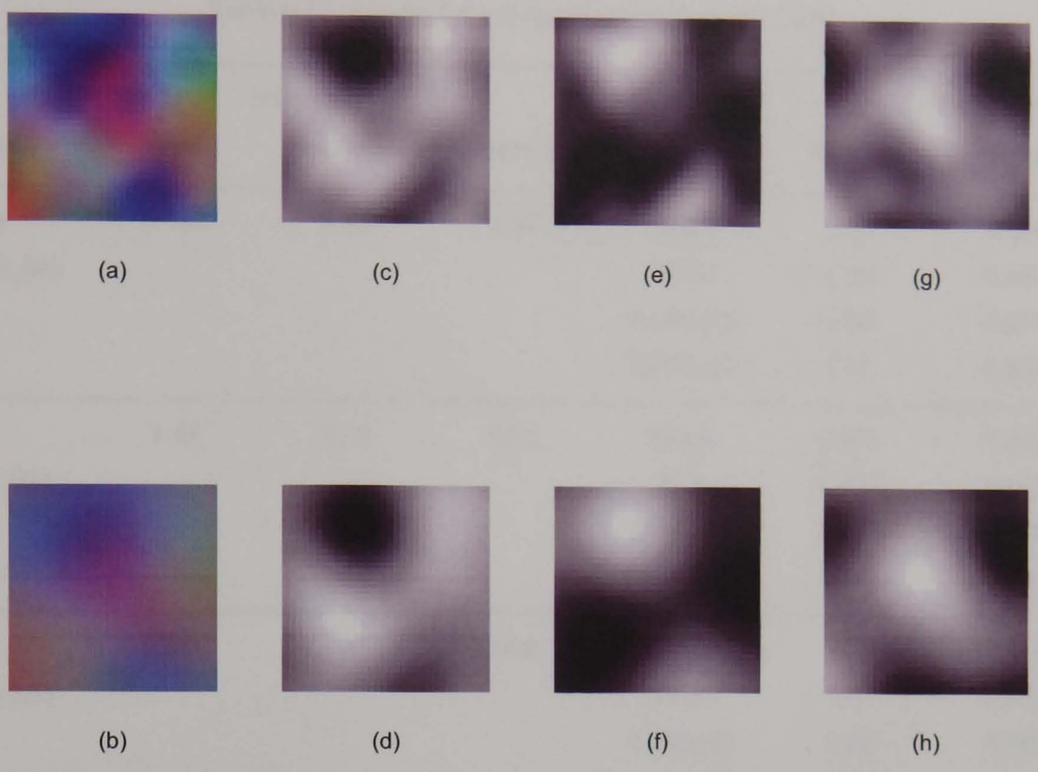


Figure 6.21: (a),(b) A 32×32 colour image segment of image 1 and 2 respectively; (c) (d) mono result; (e) (f) PCA; (g) (h) LCM (2)

For the particular segment shown in Figure 6.21 the monochrome and PCA algorithms under-estimated the depth by 8mm and 14 mm respectively. The depth produced using LCM was exact.

Further simulation experiments were performed using real textures instead of the tiled pattern created using the Genetic Algorithm. The results in Table 6.21 were formed using real textures and the only noise present in the simulations was quantisation noise. The MSEs are shown with and without the post-processing step of median filtering. The simulated scene had 10 steps with depths ranging from 0.42m to 0.62m. The spread in the HSI components are also shown.

Table 6.21. Results for sub-optimum textures for LCM

Texture	Variance / 10^{-3}			Algorithm	MSE / mm^2	
	Hue	Saturation	Intensity		Without	With
Carpet (carpet_01)	0.793	8.55	1.37	Mono	1.35	0.460
				PCA	1.36	0.467
				LCM (1)	1.64	0.657
				LCM (2)	1.50	0.568
Grass (grass_01)	1.49	32.9	24.1	Mono	0.675	0.250
				PCA	0.658	0.252
				LCM (1)	0.378	0.187
				LCM (2)	0.409	0.207
Stone (stone_03)	0.748	3.31	14.6	Mono	1.68	0.571
				PCA	1.67	0.553
				LCM (1)	2.52	0.885
				LCM (2)	2.17	0.797
Stone (stone_09)	18.5	10.4	16.1	Mono	1.34	0.462
				PCA	1.32	0.442
				LCM (1)	1.47	0.466
				LCM (2)	1.45	0.472
Wood (wood_03)	0.0650	0.702	0.862	Mono	3.92	1.56
				PCA	3.83	1.49
				LCM (1)	5.16	2.21
				LCM (2)	5.03	2.19

As discussed in Section 5.3.4 it is essentially the hue and saturation components that are important in colour mixing. The wood texture employed has very little variation in the HSI components and consequently there is very little texture, hence the poor results using the monochrome and PCA algorithms in comparison with the other four textures. Both versions of the LCM algorithms produced much worse results for all textures except the grass and this is believed to be due to the lack of the colour variation and the fact that natural textures do not possess the correct colour pertaining to aid localisation. The stone texture (stone_03) produced one of the worst results and the median filtered depth maps are shown in Figure 6.22 and Figure 6.23.

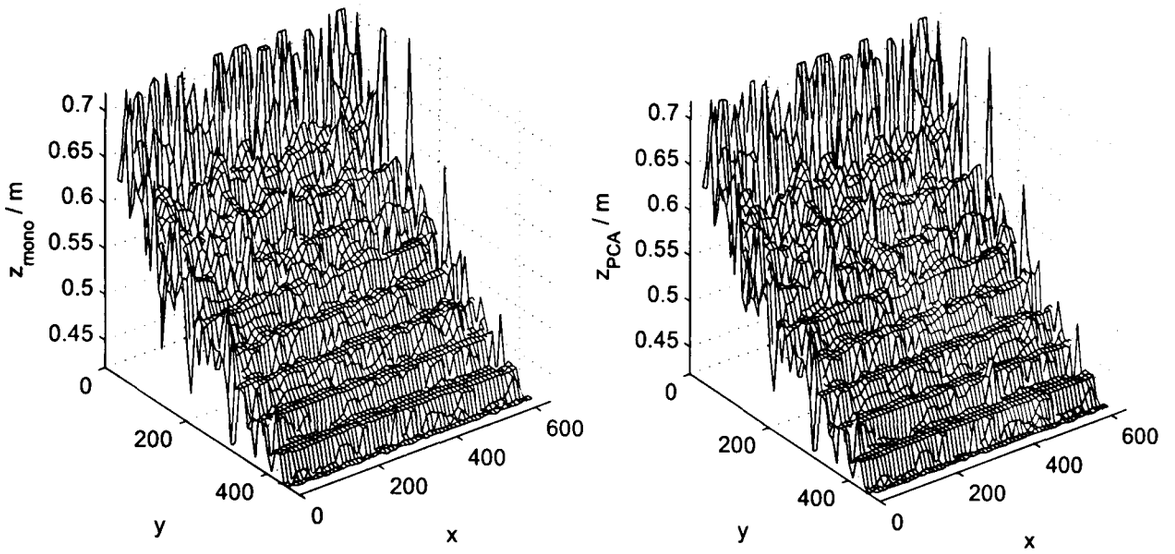


Figure 6.22: Depth map using the stone (stone_03) texture and the monochrome and PCA algorithms

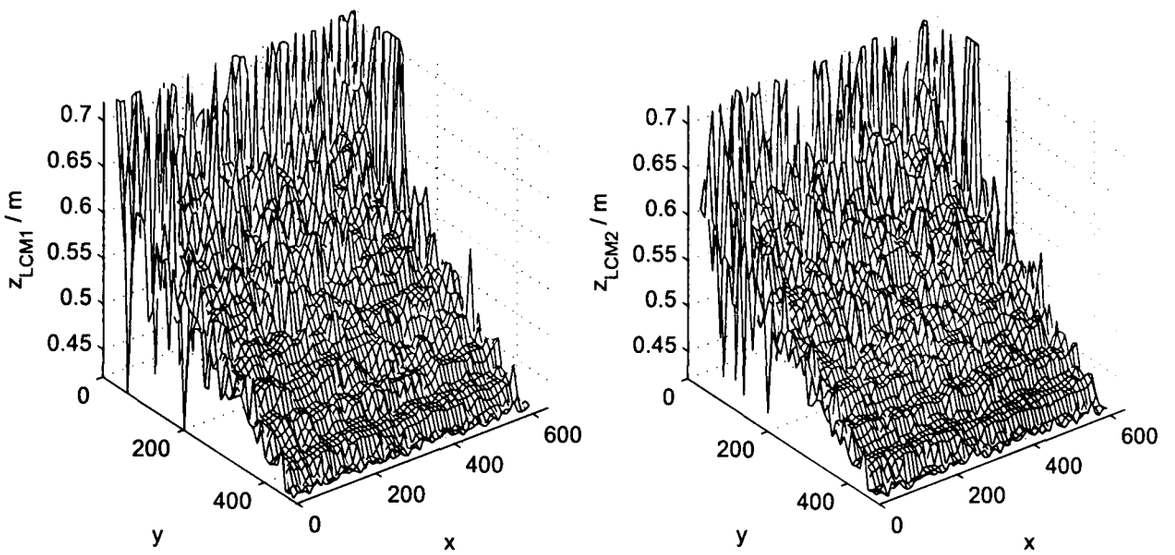


Figure 6.23: Depth map using the stone (stone_03) texture and algorithms LCM (1) and LCM (2)

The only experiment where the LCM algorithm out-performed the monochrome and PCA algorithms occurred using the grass texture, which possessed the most variation in the HSI components. The depth maps are plotted in Figure 6.24 and Figure 6.25 for comparison, where median filtering has been applied.

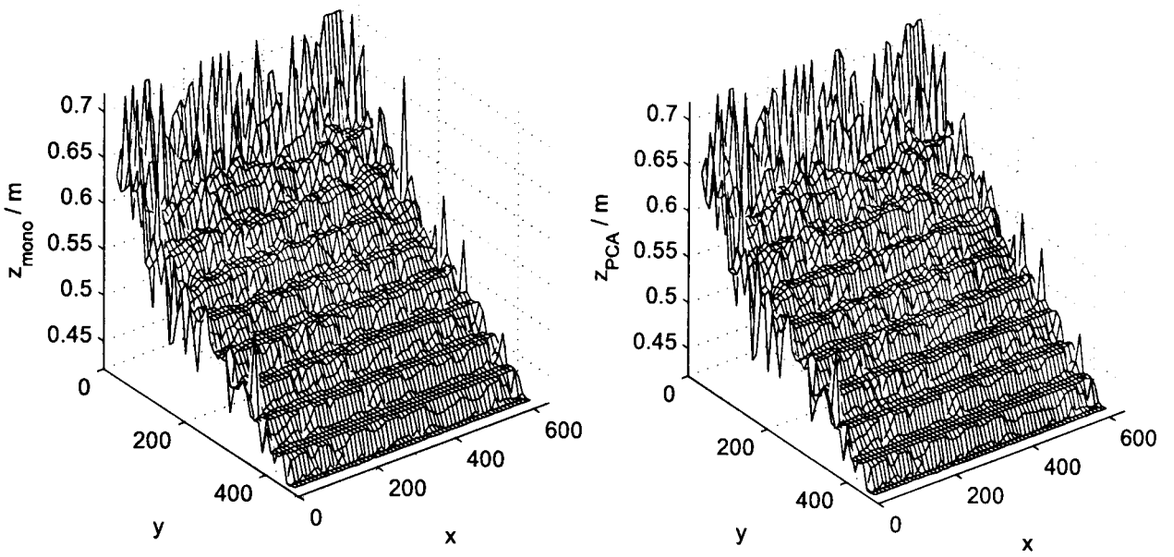


Figure 6.24: Depth map using the grass texture and the monochrome and PCA algorithms

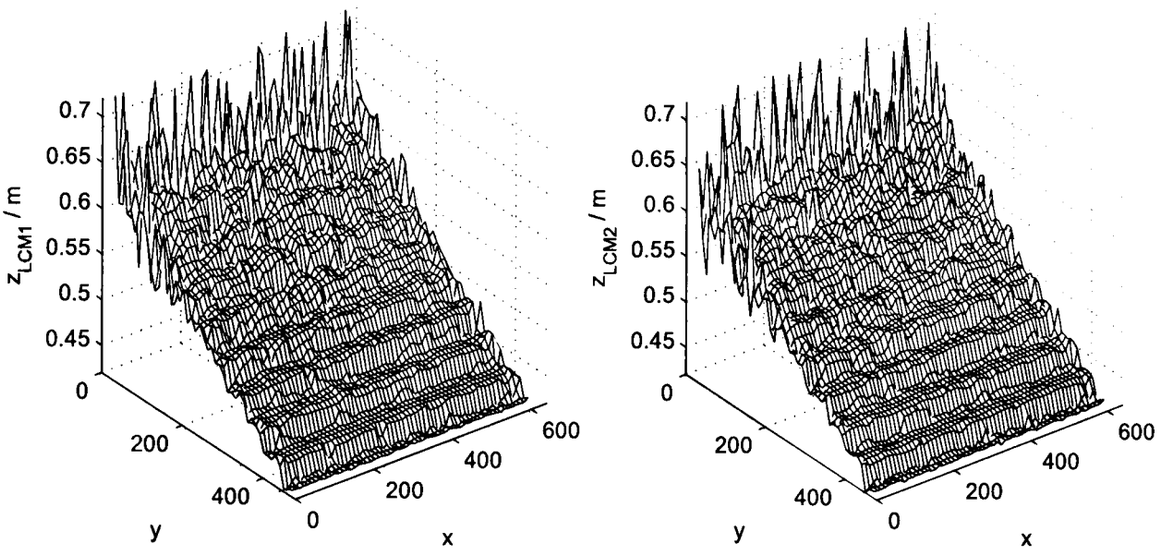


Figure 6.25: Depth map using the grass texture and algorithms LCM (1) and LCM (2)

The errors at larger depths are worse using the grass texture compared to using the optimised texture, as can be seen by comparing Figure 6.20 and Figure 6.25. However, note that the depth map at the edges of the image have been improved using LCM, even though the texture was sub-optimum.

6.7.3 Conclusion

The results showed that the best image to use to determine the optimum weightings (α, β, γ) is either image 1 or 2, depending on the scene. There is clearly a trade-off between localisation and SNR and in the experiments at around 30dB LCM and PCA performed about the same. Another formulation of LCM may lead to better results in the presence of noise, but it seems likely that the trade-off between localisation and SNR is inevitable.

6.8 Conclusion

This chapter has examined the different colour mixing algorithms that were discussed in the previous chapter and firstly the results showed that using a $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$ combination of the colour planes was not optimum. Each of the pre-processing algorithms were tested individually on a pair of defocused images where 2,745 depth estimates were measured. The mean processing time for each algorithm is summarised in Table 6.22.

Table 6.22. Comparison of the processing times

Algorithm	Mean processing time for a single 32×32 window / seconds
Monochrome	0.14
GA (with a known depth)	13
Principal Component Analysis	0.23
Maximisation of the Signal-to-Noise Ratio	2.1
Maximisation of the Fractal Dimension	3.4
Localisation through Colour Mixing	0.16

The monochrome algorithm where $(\alpha, \beta, \gamma) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$ is the fastest because of the simple pre-processing just requires an average of the colour planes. The GA with a known depth is the slowest because the GA requires 10 individuals that are evolved for 10 generations, thus for a given window 100 times more convolutions are required.

To maximise the SNR, 100 individuals were evolved for 50 generations, but the objective function is the calculation of the SNR, thus making it much quicker than requiring convolutions. It was found that 50 individuals evolved for 50 generations was sufficient to maximise the FD. The least squares fitting of the FD is computationally more intensive than the SNR, as shown by the values in Table 6.22.

The LCM algorithm used a deterministic linear algebra approach to calculate (α, β, γ) and it requires the calculation of large matrix multiplications and inverses of size $32^2 \times 32^2$. However, MATLAB is efficient at performing the required matrix calculations and the LCM algorithm is only 0.02s slower than the monochrome, whereas the PCA algorithm is 0.09s slower; which can be attributed to the calculation of the covariance matrix and eigenvalues and eigenvectors.

The GA that finds the optimum (α, β, γ) for a known depth was found to perform noise reduction, but also it could, on occasion, use the noise present in an image to ensure the DFD algorithm gave the wrong depth. It was purely of academic interest, but it clearly showed that the DFD algorithm was sensitive to the scaling constants used.

Principal Component Analysis was explored and it was found that the first component formed from the eigenvector of the major axis of a hyperellipsoid gave depth maps with lower errors than using equal weighting. In the presence of AWGN, PCA was found to be between 1.3 and 1.5 times better than using the monochrome image. The experiments in which the real camera imaged slopes with textures printed on them showed that PCA worked worse than the monochrome approach three fifths of the time. An analysis of the camera noise was performed and it was discovered that the camera had a strong multiplicative component. PCA worked well in the presence of additive noise, but was found to be adversely affected by multiplicative noise. A weighted PCA was used, called NVA-PCA, but this was not found to be a successful solution. The RGB colour plane with the largest variance generally gave the minimum depth error as would be expected, but clearly a new formulation of PCA is required that is robust in the presence of multiplicative noise.

An algorithm was developed that maximised the SNR of an image, assuming an additive noise model, by colour mixing. Due to the results of the noise experiments on the actual camera it was shown that the algorithm could not be tested practically. In simulation it was found that PCA and maximising the SNR gave essentially the same results when the noise variance of each colour plane was the same, and this was explained by the theoretical work of Rosipal *et al.* [186]. When the noise is uncorrelated and has a different variance in each plane (i.e. it is non-isotropic), it was found that the algorithm to maximise the SNR increased the SNR by between 2.1dB and 2.3dB compared to monochrome and by 3.0dB to 3.5dB compared to using PCA. The resulting MSEs showed improvements of between 3.4 and 7.8 times compared to using monochrome and between 1.7 to 1.9 using PCA.

The fractal dimension of a monochrome image gives a measure of the texture variation and it was found that maximising the FD using colour mixing based on modelling the image using fBm reduced the SNR. The consequence of reducing the SNR was that the depth maps were worse than both the monochrome and PCA algorithms. A solution to

this problem may be to use a multi-objective optimisation algorithm to boost the texture and reduce the noise of image I, for example using

$$\max_{(\alpha,\beta,\gamma)} (\lambda_1 \text{FD}[I] + \lambda_2 \text{SNR}[I]) \tag{6.15}$$

where λ_1 and λ_2 are adjusted to give the appropriate weighting. However, as PCA essentially performs this task there may be little merit in the approach.

The LCM algorithm was designed to reduce the image overlap and windowing problems and it is an active DFD technique that would require a data or slide projector. The optimum projected pattern was evolved using a GA and it was found to be a random coloured pattern. For a given image region, the scaling constants (α, β, γ) were determined using the Moore-Penrose pseudo-inverse. With an image composed of 10 steps spanning a depth range of 0.42m to 0.62m it was found that the MSE of the depth map using LCM was between 1.7 and 2.2 times better than PCA and between 7.3 and 9.4 times better than monochrome. The trade-off between SNR and localisation was found, but in practice it did not hinder the results. In its current formulation, LCM requires SNRs of 30dB or greater to achieve a significant improvement, but a multi-objective approach could again be considered that accounts for the localisation and the SNR to ensure a usable trade-off.

Chapter 7

Image Normalisation for Depth-From-Defocus

7.1 Introduction

In this, the penultimate chapter, the improvement to the image normalisation that was discovered during the final stages of the research is presented. Ens and Lawrence's [58] [59] DFD algorithm required two images to be taken with different camera parameters. The aperture size (f-number) was changed between images to ensure no image spatial registration problems, but with the consequence that for a given exposure time, the relative image brightnesses were different. The problem of the normalisation of the images taken with different apertures was examined from theoretical and experimental perspectives.

In Section 7.2 image formation is considered and the effect of the f-number is analysed. Experimental results of changing the f-number are presented in Section 7.3 for the 24mm Sigma photographic lens. The PCA algorithm from the previous chapter was tested against the monochrome algorithm to show whether the normalisation problems resulted in poorer depth maps using colour mixing. An analysis of the effect of colour on depth accuracy is presented in Section 7.4. The DFD results of using more complex scenes than were employed in the previous experiments are shown in Section 7.5 and then the conclusions drawn in Section 7.6.

7.2 Theoretical Analysis

7.2.1 Introduction

Ens and Lawrence's [58] [59] DFD algorithm searches for the optimum convolution ratio $h_3(x, y)$ such that

$$\min \sum_{x,y} (i_1(x, y) * h_3(x, y) - i_2(x, y))^2 \tag{7.1}$$

where $i_1(x, y)$ and $i_2(x, y)$ are the images taken with the first and second set of camera parameters respectively. In this research, the f-number was changed between images to ensure no image registration problems. The mechanical construction of the 24mm Sigma lens meant that the f-number snaps into position, and thus it was safely assumed that the same parameters existed each time.

Image 1 was taken with a smaller aperture than image 2 resulting in a sharper image (i.e. less defocused), but also a lower brightness for a given exposure time. It was important that the images were scaled to compensate for the change in brightness and the problem becomes searching for $h_3(x, y)$ using

$$\min \sum_{x,y} (i_{1_N}(x, y) * h_3(x, y) - i_{2_N}(x, y))^2 \tag{7.2}$$

where $i_{1_N}(x, y)$ and $i_{2_N}(x, y)$ are the brightness normalised versions of images 1 and 2 respectively. For convenience the result of the convolution $i_{1_N}(x, y) * h_3(x, y)$ is denoted $\hat{i}_{2_N}(x, y)$.

7.2.2 Statistical Normalisation Approach

Ens and Lawrence's DFD algorithm was described in Section 5.2 and the problem of normalising the more defocused image $i_2(x, y)$ and the less defocused image $i_1(x, y)$ convolved with the convolution ratio $\hat{i}_2(x, y) = i_1(x, y) * h_3(x, y)$ was discussed. The initial solution was to normalise such that the image segments had intensity values that lie in the closed interval [0, 1] using

$$i_{2_N}(x, y) = \frac{i_2(x, y) - \min[i_2(x, y)]}{\max[i_2(x, y)] - \min[i_2(x, y)]} \tag{7.3}$$

$$\hat{i}_{2_N}(x, y) = \frac{\hat{i}_2(x, y) - \min[\hat{i}_2(x, y)]}{\max[\hat{i}_2(x, y)] - \min[\hat{i}_2(x, y)]}. \tag{7.4}$$

The problem with the idea is that it is solely dependent on the outliers, i.e. the minimum and maximum values in the image segments, and thus it is very sensitive to noise.

A different solution was proposed that uses the statistics of the image segments that are based on all of the pixels and not just on the two outliers. The image segments $i_2(x, y)$ and $\hat{i}_2(x, y)$ were normalised using

$$i_{2_N}(x, y) = \frac{i_2(x, y) - E[i_2(x, y)]}{\sqrt{\mathrm{Var}[i_2(x, y) - E[i_2(x, y)]]}} \tag{7.5}$$

$$\hat{i}_{2_N}(x, y) = \frac{\hat{i}_2(x, y) - E\left[\hat{i}_2(x, y)\right]}{\sqrt{\mathrm{Var}\left[\hat{i}_2(x, y) - E\left[\hat{i}_2(x, y)\right]\right]}} \tag{7.6}$$

where $E[X]$ and $\mathrm{Var}[X]$ denote the expected value and variance of X respectively and so i_{2_N} and $\hat{i}_{2_N}$ have zero mean and unit variance.

7.2.3 Radiance Analysis

Consider a point on an object S that has a radiance of L $(W\,m^{-2}\,\mathrm{srad}^{-1})$ and is being imaged by a lens system, as shown in Figure 7.1. The lens has a focal length of F and a transmittance T. The distance between the image plane (i.e. the CCD) and the lens is denoted v (m) and the principal ray makes an angle θ (rads) with the optical axis. The aperture is assumed to be circular with a diameter d (m). If the lens is focused then an image of S, denoted S', will be produced on the screen at I_F.

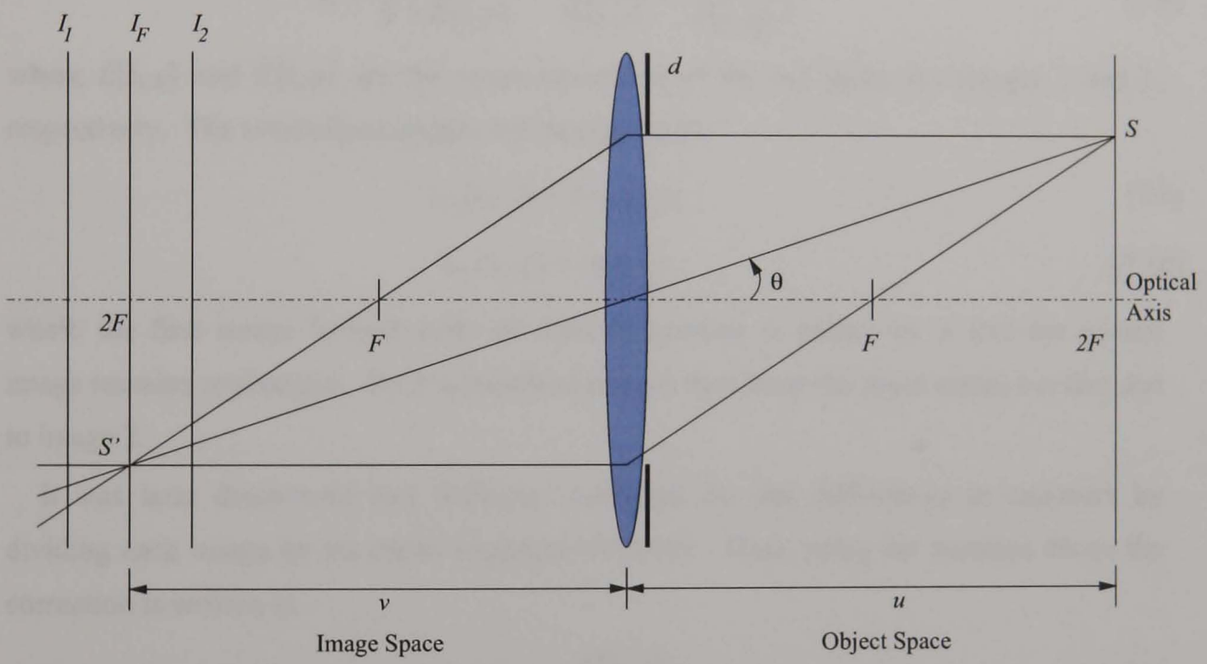


Figure 7.1: Diagram of a focused lens system

It can be shown that the irradiance on the screen or CCD due to point S is given by [132]

$$E = T\,L\left(\frac{\pi\,d^2}{4}\right)\frac{\cos^4\theta}{v^2}. \tag{7.7}$$

Suppose the diameter of the lens is reduced by $\frac{1}{\sqrt{2}}$ then the irradiance becomes $\frac{E}{2}$ using (7.7), i.e. the image is half as bright. Changing the aperture by one f-stop corresponds to a halving or doubling of the image intensity depending on direction. The 24mm Sigma photographic lens can be set to half f-stops, and thus the diameter reduces by $\frac{1}{\sqrt{2}}$ and the irradiance changes by $\frac{E}{\sqrt{2}}$.

Equation 7.7 is for a focused lens system and for a defocused system the energy is spread out. For example, if the image plane is at I_1 or I_2 then point S is no longer imaged to the conjugate point S'. Due to the conservation of energy, the radiance of the point decreases. The shape is determined by the PSF and as described in Chapter 3, assuming geometrical optics the shape is a pillbox. For convenience in finding the convolution ratio, the Gaussian shape was assumed.

7.2.4 Actual Radiance Analysis

If the aperture is reduced by one or two stops then the image intensity should decrease to a half or a quarter respectively. It is unlikely that an optical system with an aperture will exactly give a halving of intensity with each f-stop, and further the ratios may be different for each colour plane. An average of the ratios is given by

$$\phi = \frac{1}{3}\left(\frac{E[i_{2R}]}{E[i_{1R}]} + \frac{E[i_{2G}]}{E[i_{1G}]} + \frac{E[i_{2B}]}{E[i_{1B}]}\right) \tag{7.8}$$

where $E[i_{1R}]$ and $E[i_{2R}]$ are the mean intensities of the red plane for images 1 and 2, respectively. The normalised images are then given by

$$i_{1_N}(x, y) = \phi\, i_1(x, y) \tag{7.9}$$

$$i_{2_N}(x, y) = i_2(x, y) \tag{7.10}$$

where the first image formed with the smaller aperture is scaled by ϕ and the second image remains unchanged. Both normalised images then have the same mean, i.e. that due to image 2.

It was later discovered that Subbarao corrected for the differences in exposure by dividing each image by the mean brightness [4] [79]. Thus, using the notation above the correction is written as

$$i_{1_N}(x, y) = \frac{i_1(x, y)}{E[i_1(x, y)]} \tag{7.11}$$

$$i_{2_N}(x, y) = \frac{i_2(x, y)}{E[i_2(x, y)]}. \tag{7.12}$$

This is essentially the same normalisation as proposed using (7.8), (7.9) and (7.10), except the mean of the images is unity, instead of having the mean brightness of image 2.

7.2.5 Conclusion

This section has discussed four possible normalisation methods. The original approach using the minimum and maximum intensities is dependent on outliers and it was expected that the statistical approach then ensures $i_{1_N}(x, y)$ and $i_{2_N}(x, y)$ have unit variance and zero mean would perform better. Theoretically, changing the f-number by one or two stops will change the intensity by a factor of 2 or 4, but it was important to test this assumption and so the next section presents experimental results.

7.3 Experimental Results

7.3.1 Introduction

In the next subsections the mean intensities of each colour plane were examined as a function of the aperture for a given exposure time and then in Section 7.3.3, DFD results are shown using each of the four normalisation approaches described previously.

7.3.2 Intensity Dependence on Aperture Results

The Basler A631fc colour camera with the 24mm Sigma photographic lens was used to image a slope with a colour checkerboard pattern pasted on to it. The plane had a distance that changed smoothly from 0.440m to 0.520m from the camera. The aperture was fully opened with an f-number of f/2.8 and then the shutter time set to the maximum value that did not incur image saturation. Twenty images for the particular aperture setting were taken and the images averaged to reduce the noise. The mean brightness of the red, green and blue planes were then calculated. Apertures up to f/16 were tested using the same procedure. Importantly, the exposure time was fixed for all images. The results of the experiment are presented in Table 7.1.

Table 7.1. Mean intensity of each colour for a set exposure time

	Mean Intensity		
Aperture	Red	Green	Blue
f/2.8	118.3	114.7	106.1
f/3.4	118.3	114.7	106.1
f/4	105.5	102.6	94.2
f/4.8	74.1	72.5	65.6
f/5.6	54.8	53.7	48.4
f/6.7	38.3	37.6	33.6
f/8	26.4	26.1	23.0
f/9.5	18.9	18.8	16.5
f/11	13.0	12.8	11.3
f/13.5	9.5	9.2	8.3
f/16	6.7	6.5	5.8

A particularly noticeable feature of the results is that there was no change in the mean intensities using apertures of f/2.8 and f/3.4 for all three colour planes. The ratio of the intensities between two apertures half an f-stop apart should be $\sqrt{2}$ and Figure 7.2 shows the actual ratios and the expected ratio (horizontal solid line). For apertures of (f/2.8, f/3.4) and (f/3.4, f/4) the ratio is much less than $\sqrt{2}$. The experiments were repeated a few times in an attempt to eliminate experimental error, however, it was later learnt that it is not uncommon for lens manufacturers to 'misquote' the fastest lens speed. Often, the maximum f-number is calculated using

$$f = \frac{F}{D} \tag{7.13}$$

where F is the focal length and D is the diameter of the front of the lens, but the equation does not take into account the effect of the diameters of the lens elements. The experiments appear to show that the 24mm Sigma lens is not a true f/2.8 in terms of light gathering capability.

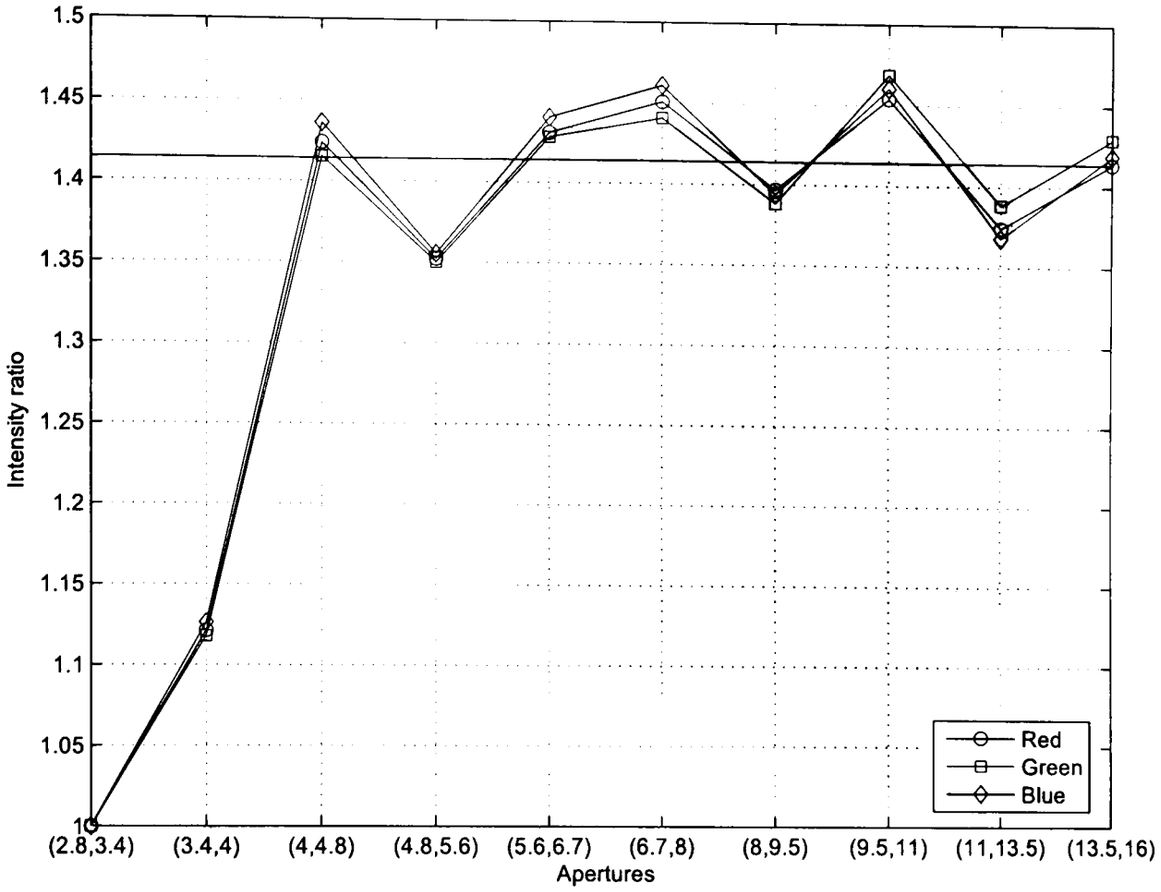


Figure 7.2: Relative brightness as a function of the aperture

The remaining eight half-stop aperture combinations performed as expected with intensity ratios of approximately $\sqrt{2}$. The results show that it is important to experimentally test the light gathering capability of the lens.

Due to the age of the lens, Sigma were unable to provide a datasheet from which information about the light gathering capability could be found and thus compared to the experimental results. However, due to the simplicity of the test and the fact that it was run a couple of times to check the result, experimental error could be safely be ruled out.

7.3.3 Depth-From-Defocus Results

The PSF of the lens was found for apertures of f/2.8, f/4 and f/5.6 and thus DFD can only be performed using those apertures. The four image normalisation ideas were tested on the images of the slope that were used in the previous section and the MSE results without median filtering are presented in Table 7.2. The colour images were converted to monochrome using an equal weighting of the colour planes, i.e. $\alpha = \beta = \gamma = \frac{1}{3}$. The sum of the L_2-norms was used as the error measure and each depth map was composed of 2745 points and took around 16 minutes to process.

Table 7.2. MSE results for the normalisation algorithms

Normalisation	MSE for a given Aperture Combination / mm^2		
	(f/5.6, f/4)	(f/5.6, f/2.8)	(f/4, f/2.8)
min-max	0.783	0.838	3.56
Statistical approach	0.474	0.652	3.67
Theoretical scaling	12.1	62.2	57.1
Actual scaling	0.404	0.767	3.68

For the aperture combinations of (f/5.6, f/4), (f/5.6, f/2.8) and (f/4, f/2.8) the actual scaling values were found to be 1.93, 2.16 and 1.12 and theoretically they should be 2, 4 and 2 respectively. The aperture combination (f/4, f/2.8) was left in for completeness, even though the MSE results were poor on the checkerboard images discussed in Section 6.2.10. Theoretically, the mean intensity of the image taken with f/2.8 should be twice of that taken with f/4, but in fact the ratio was only 1.12, and consequently the MSE results using the theoretical value are much worse than the other methods.

For the two usable aperture combinations, (f/5.6, f/4) and (f/5.6, f/2.8), the statistical approach to normalisation reduced the MSE by 1.7 and 1.3 times respectively. The results using the theoretical scaling were the worst of the set. Using the actual scaling resulted in the best MSE of the possible algorithms for (f/5.6, f/4), but with (f/5.6, f/2.8) it was outperformed by the statistical approach. The depth maps using the statistical normalisation approach and scaling by the actual values are shown in Figure 7.3 and Figure 7.4.

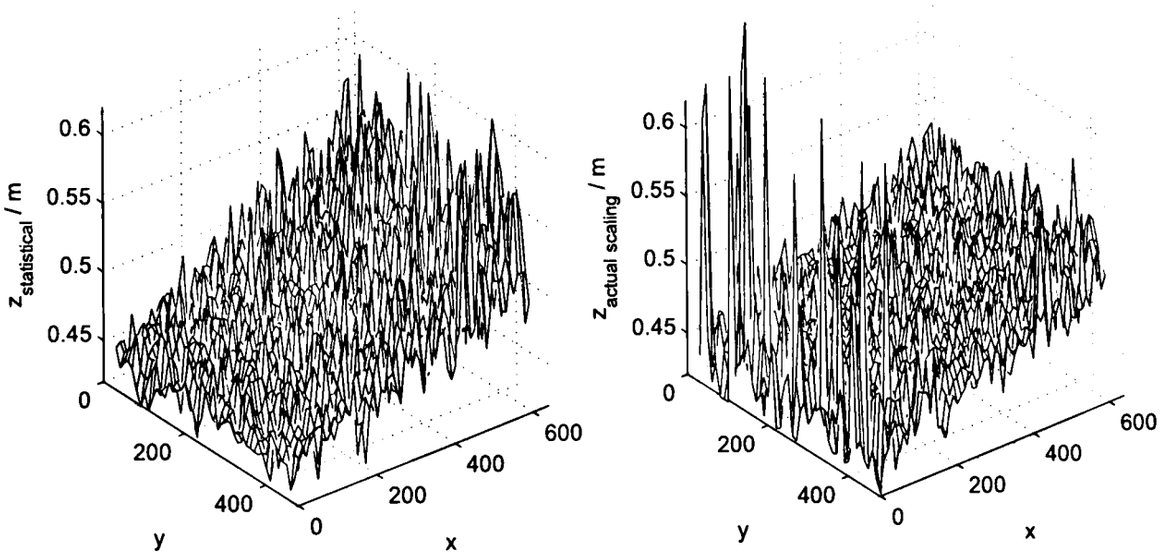


Figure 7.3: Depth maps using f/5.6 and f/4 using the statistical-based normalisation (left) and the experimentally determined scaling (right)

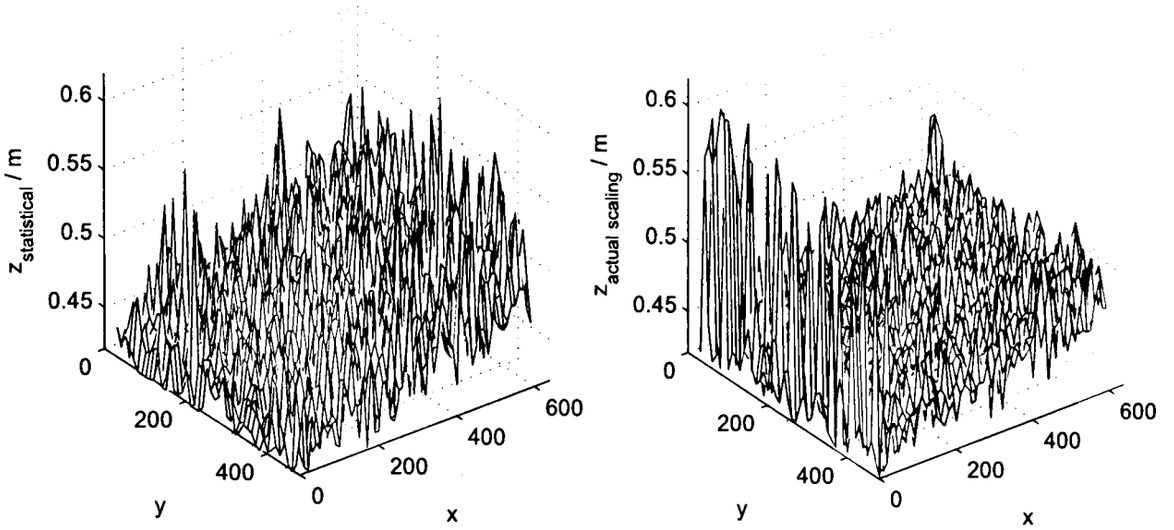


Figure 7.4: Depth maps using f/5.6 and f/2.8 using the statistical-based normalisation (left) and the experimentally determined scaling (right)

When the slope is close to the camera, i.e. a depth of around 0.44m, the depth is massively over-estimated by the DFD algorithm when the image normalisation was performed using the experimentally-derived scaling constant. Unfortunately, a reason for this was unknown.

7.3.4 Statistical Normalisation Results

To illustrate the new normalisation, the images captured using the colour camera and the 24mm Sigma lens that were used to test the PCA algorithm were re-tested. The results are given in Table 7.3 and for ease of comparison only the MSE results are presented.

Table 7.3. Results using the different normalisation algorithms

Texture	Algorithm	Statistical Normalisation		Min-Max Normalisation	
		MSE w/o MF / $10^{-3}m^2$	MSE w MF / $10^{-3}m^2$	MSE w/o MF / 10^{-3} m^2	MSE w MF / $10^{-3}m^2$
Carpet (carpet_01)	Mono	0.0978	0.0463	0.202	0.0755
	PCA	0.122	0.0631	0.239	0.0832
Colour Checkerboard	Mono	0.259	0.127	0.456	0.210
	PCA	0.160	0.0717	0.320	0.141
Grass (grass_02)	Mono	0.108	0.0480	0.261	0.0958
	PCA	0.118	0.0519	0.286	0.104
Stone (stone_03)	Mono	0.657	0.200	1.02	0.295
	PCA	0.682	0.209	1.08	0.322
Stone (stone_08)	Mono	0.127	0.0888	0.253	0.150
	PCA	0.108	0.0647	0.230	0.122

The results in Table 7.3 show that by using the new normalisation based on the image statistics that the MSE has been nearly halved in comparison to the old method. The results using PCA are only better than the monochrome case using the colour checker-board and the stone (stone_08) texture, as with the original normalisation. Thus, it appears as though the relative accuracy compared to monochrome algorithm was not dependent on the poor normaliation used in the results of the previous chapter.

7.4 The Effect of Colour on Depth Accuracy

7.4.1 Introduction

This section considers whether the performance of the DFD algorithm is influenced by the colour of the objects in the scene, even though the images are converted to monochrome using an equal weighting of the colour planes. It was shown in Section 6.4.3 that the noise is predominantly multiplicative and further the green plane has the lowest noise variance due to being sampled twice as much as the red and blue planes because of the Bayer filter employed. The blue colour plane has the largest noise variance and this was attributed to decreasing sensitivity of the semiconductor-based CCD as the wavelengths extend beyond the peak sensitivity in the infrared region. The reduced sensitivity to blue light means that a higher gain is required, thus resulting in a higher noise level.

7.4.2 Theoretical Analysis

Consider a colour camera imaging a scene that has a brightness variation where the underlying probability density function of the texture is a Gaussian with a mean μ and a standard deviation σ. For DFD work it is important that the camera's output is not saturated and simultaneously that it is above the clamping level. This ensures that the output of the camera is a linear function of the scene brightness.

Consider a camera with an 8-bit ADC (thus giving intensity values in the range 0 to 255) imaging a texture with a mean $\mu = 180$ and a standard deviation $\sigma = 35$. The quantisation of the number of photoelectrons in each pixel means that intensities fall in discrete bins. If the camera saturates then the discrete PDF will take the form shown in Figure 7.5. Note how the saturation has caused the bin corresponding to an intensity of 255 to have a higher probability than it should when compared to the limiting distribution.

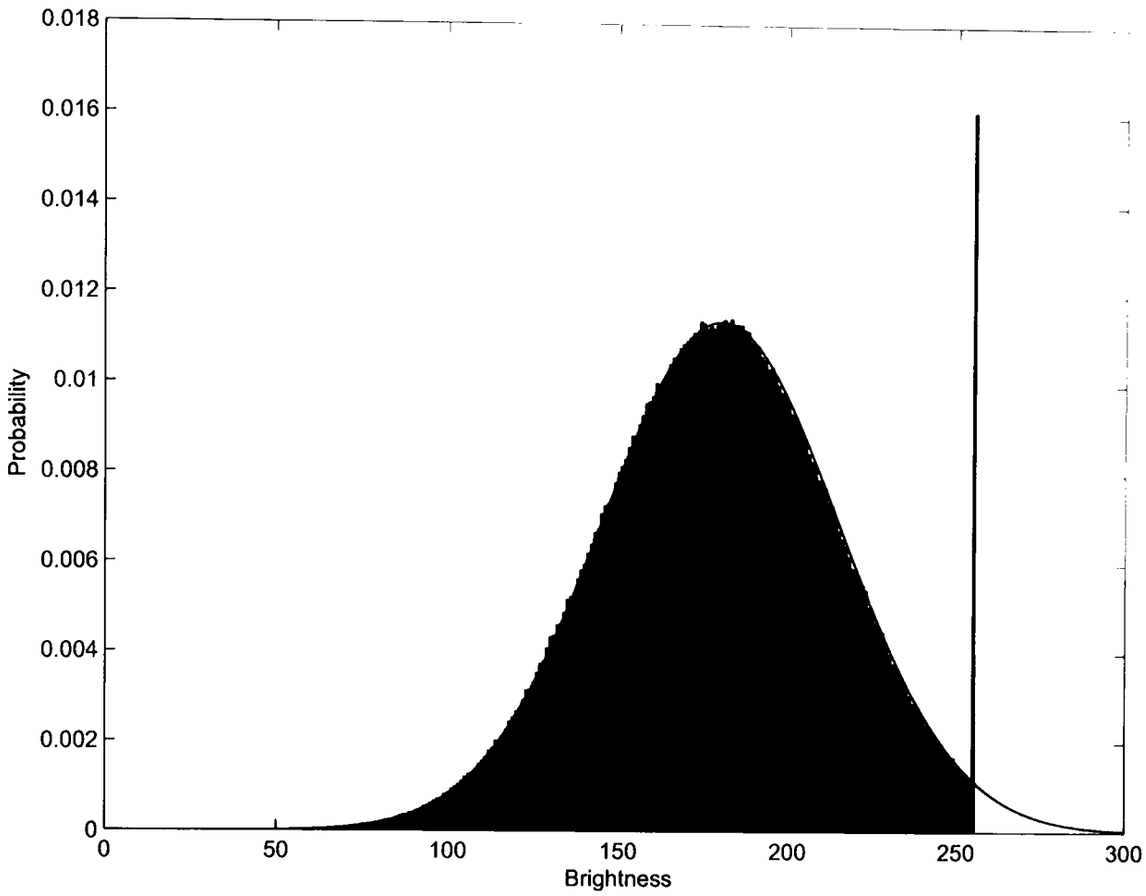


Figure 7.5: The limiting distribution of the brightness of the scene (solid line) and a histogram of intensities as produced by the camera (bars)

The inequality describing the allowable region of the means and variances of the texture for the response to be linear is given by

$$(i_L \leq \mu - \eta\sigma) \bigwedge (i_U \geq \mu + \eta\sigma) \bigwedge (\eta\sigma \geq 0) \tag{7.14}$$

where i_L and i_U are the lower and upper intensities of the camera and η is the number of standard deviations that the texture is assumed to cover. The Gaussian was taken to have an extent of $\mu \pm 3\sigma$, and so $\eta = 3$, as the probability of an intensity occurring outside this range is very small at only 0.26%. The upper level was taken as 255 because the camera had an 8-bit ADC and the lower level was taken as 31 as this was the mode offset of the three colour planes for the Basler camera. Figure 7.6 shows the allowable region given the parameters. It should be noted that the use of the Gaussian PDF was for convenience and because it is realistic, however, the same analysis could be performed for any underlying PDF.

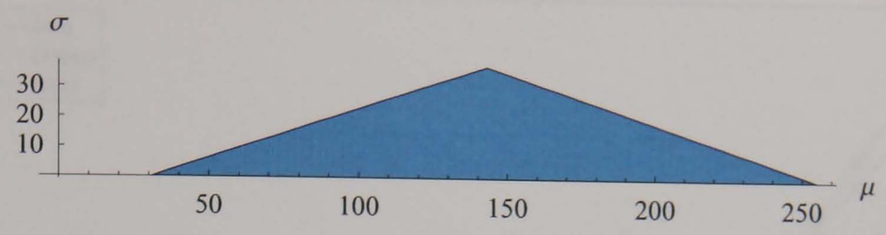


Figure 7.6: Allowable mean and standard deviations exist inside the shaded region

From the graph it can be seen that the maximum texture variance of $\sigma = 37.3$ can exist when the mean brightness is half way between the lowest and highest bounds, which in the example is $31 + (255 - 31)/2 = 143$.

The statistical normalisation performed on the images used for DFD and discussed in Section 7.2 removes the brightness variation between images and sets the mean to zero. Clearly, a textureless scene is useless for DFD as there is no defocus information, regardless of whether the scene is bright or dark. Thus, it is brightness variation that is important. However, as just discussed, the mean brightness dictates the range of allowable variance (or standard deviation) of the texture such that the camera's response remains linear.

This theoretical analysis shows that the depth accuracy will be a function of the brightness of the texture to an extent as it in turns dictates the allowable variance. The multiplicative noise variance was different for each colour plane suggesting a dependence on colour.

7.4.3 Experimental Results

In order to test how the colour, mean and variance of the texture affects the depth map accuracy, patches of texture with intensities governed by the Gaussian distribution were created with the required mean and standard deviation. In the initial tests the textures were printed on a colour laser printer, as used in the previous Chapter's experiments, but it was soon discovered that many tests would need to be performed to ascertain the correct mean and standard deviation of the texture. Also, the available colour printer's response to a blue-only texture was poor and so a TFT laptop screen was employed to provide texture. The image on the screen could then be changed quickly and the experiments performed with ease. Tests with constant intensity patches (see Figure 7.7) showed that the screen is gamma corrected, but this could not be turned off. The effect was to stretch the input intensity towards the brighter end of the response. The textures generated in MATLAB were not gamma corrected for the TFT screen, as the actual distribution of intensities was not important.

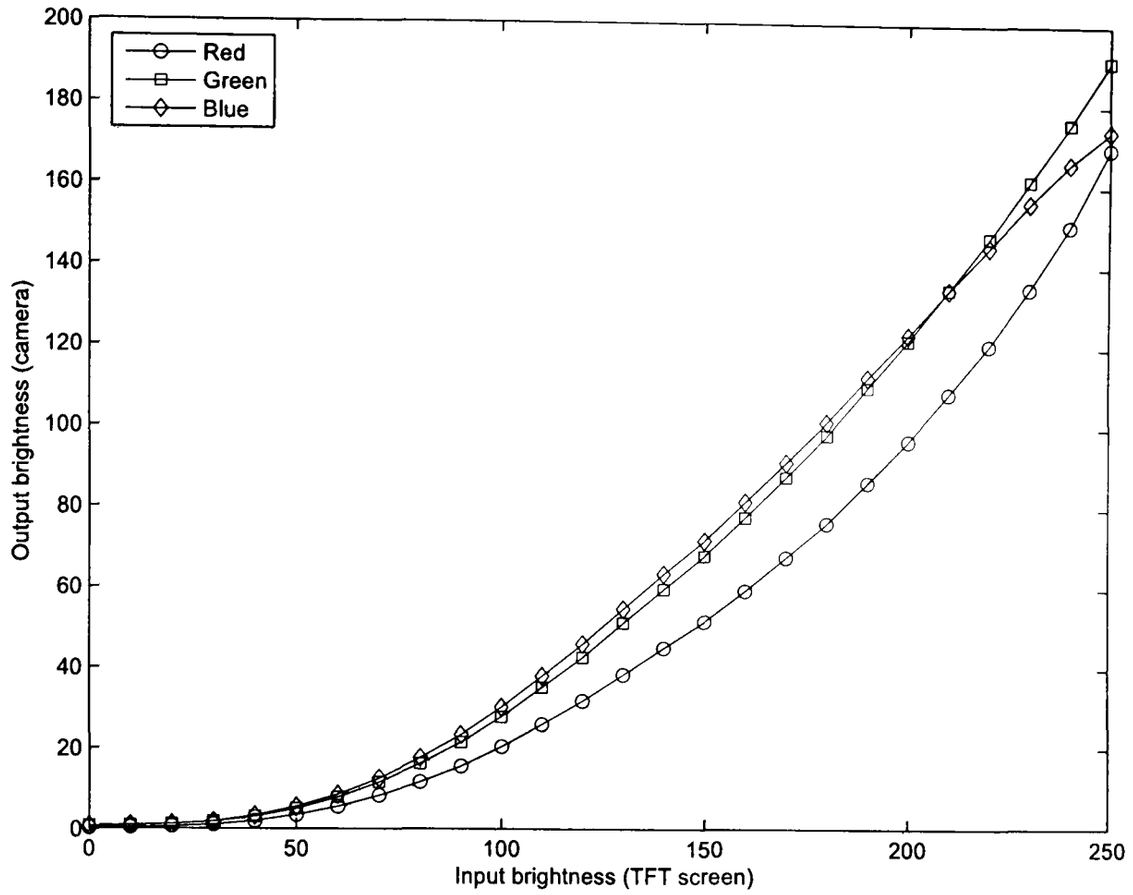


Figure 7.7: Response of the camera to the input brightness on the TFT screen

Red-only, green-only and blue-only textures were displayed on the TFT screen 0.460m from the camera. Each pattern was composed of nine squares, each with a specific mean brightness and standard deviation, an example of which is shown in Figure 7.8. Eight images for a given aperture (f/5.6 and f/2.8) were averaged to reduce noise. The colour images were then converted to monochrome using equal weightings and then processed by the implementation of Ens and Lawrence's DFD algorithm using experimentally determined PSF measurements.

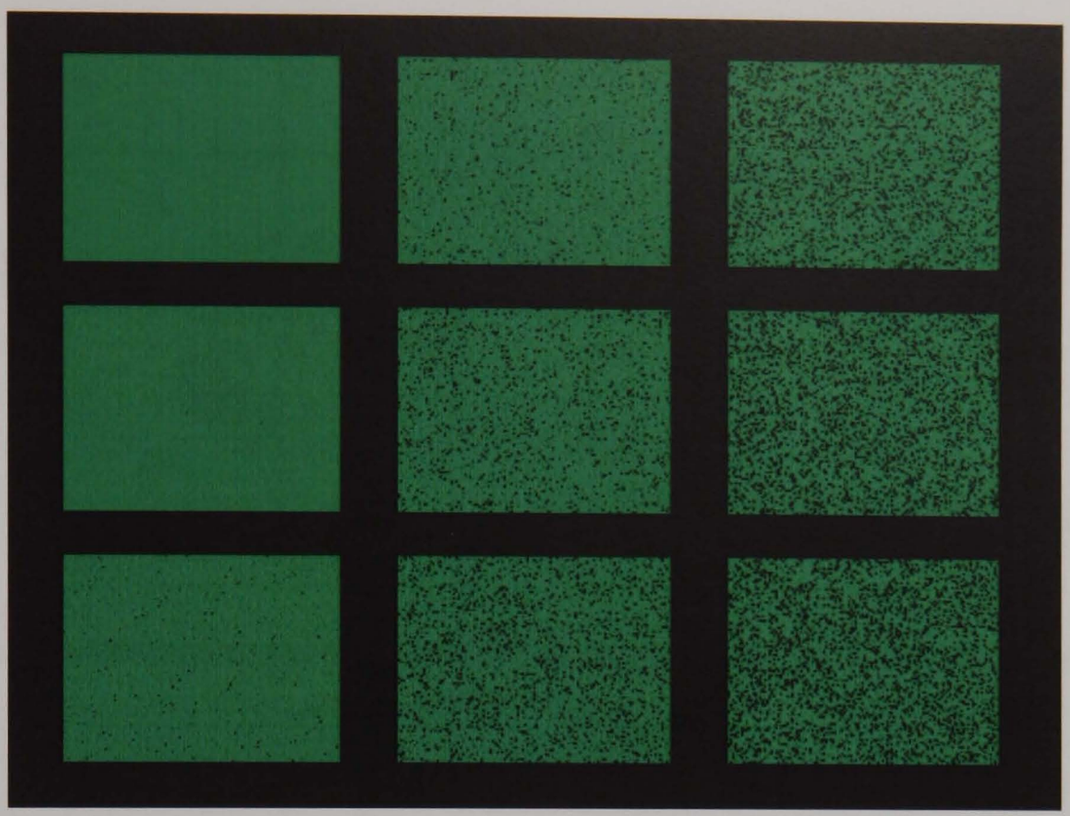


Figure 7.8: An example of the texture used in the experiments

Once all of the test images had been processed, the mean MSE of the depth for a patch was plotted as a function of the brightness and variance of the patch. The results from the red, green and blue patches are shown in Figures 7.9 to 7.11 along with histograms of the intensities for each colour plane.

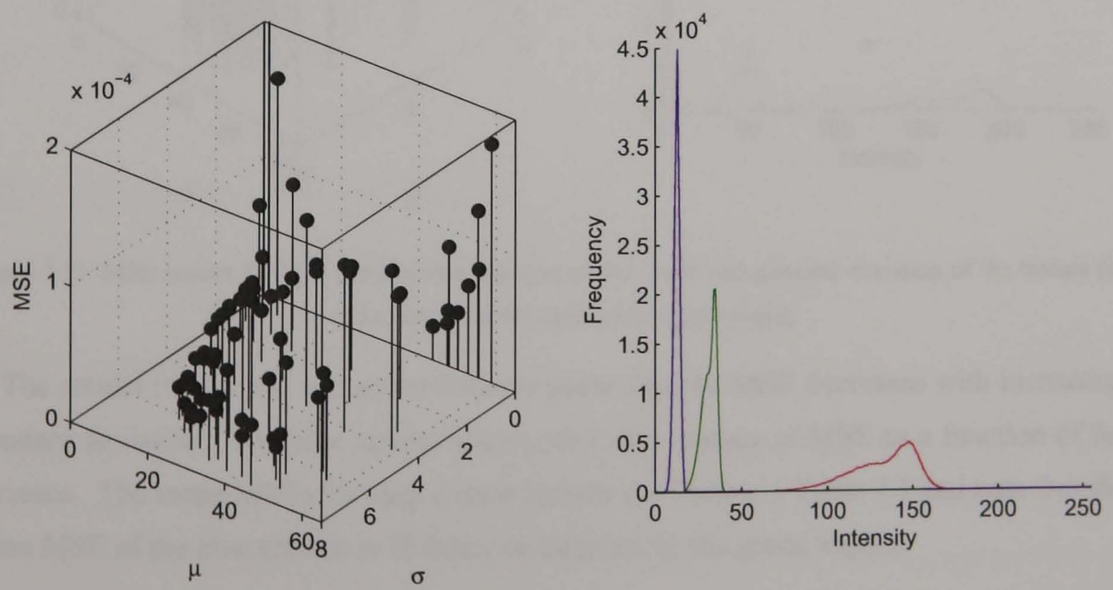


Figure 7.9: MSE results for red patches as a function of the mean and standard deviation of the texture (left) and a histogram for each colour plane (right)

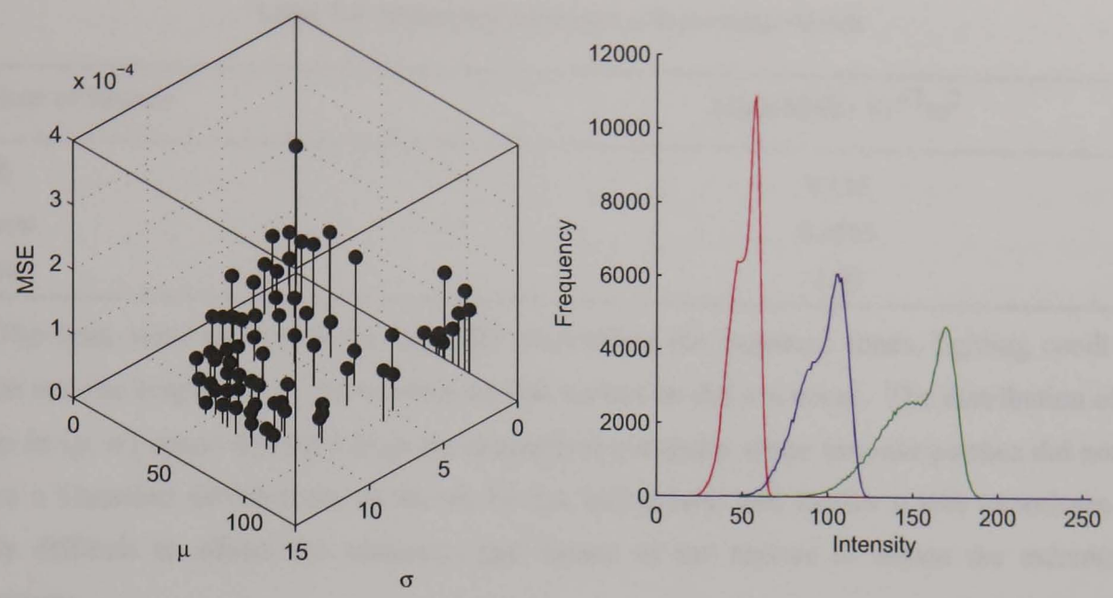


Figure 7.10: MSE results for green patches as a function of the mean and standard deviation of the texture (left) and a histogram for each colour plane (right)

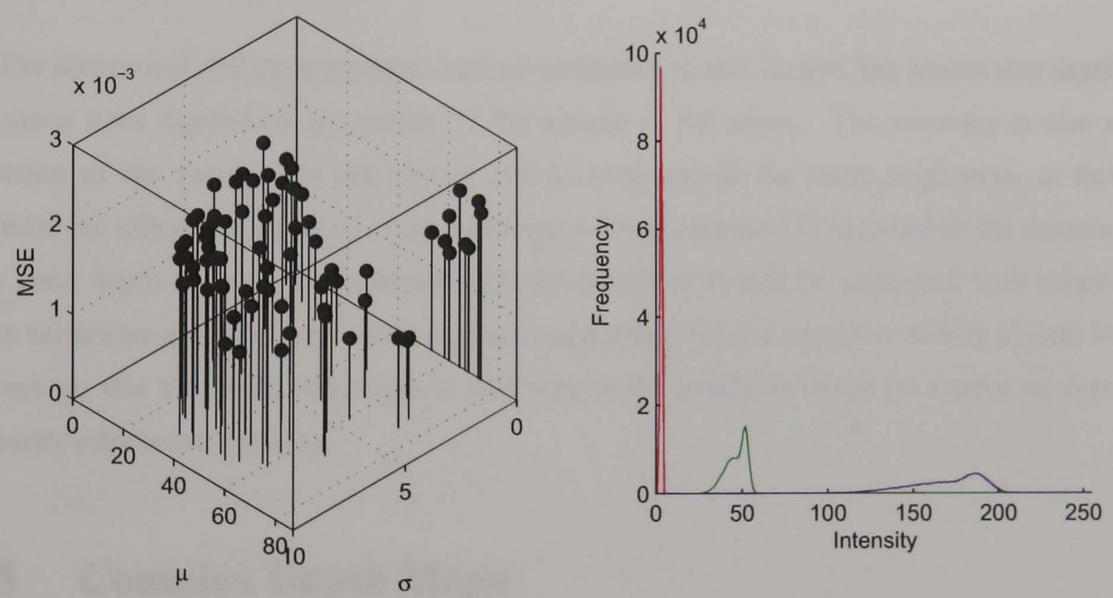


Figure 7.11: MSE results for blue patches as a function of the mean and standard deviation of the texture (left) and a histogram for each colour plane (right)

The results for the red and green textures show that the MSE decreases with increasing standard deviation. The blue texture shows very little change of MSE as a function of the variance. The mean MSEs for each colour texture are shown in Table 7.4 and note that the mean MSE of the blue texture is 28 times as large as for the green texture.

Table 7.4. Mean MSEs for each colour texture tested

Colour of Texture	Mean MSE / $10^{-3}m^2$
Red	0.115
Green	0.0563
Blue	1.60

The tests were performed by carefully controlling the exposure times, lighting conditions and the brightness of the textures so that saturation did not occur. The distribution of tests in (μ, σ) space deviated from the theoretical triangular shape because patches did not have a Gaussian distribution, as shown by the histograms, and further it was experimentally difficult to adjust the variances and means of the texture to obtain the extreme positions.

7.4.4 Conclusion

The theoretical and experimental analysis presented in this section has shown that depth accuracy does depend on the colour of the texture in the scene. The accuracy is also a function of the variance of the texture and to some extent the mean brightness, as this dictates the allowable variance range assuming a linear response is required of the camera. For good depth accuracy, a predominantly green texture should be employed with brightness variations giving a high variance, thus suggesting that the exposure setting should be chosen so that the mean brightness is half way in the available range (to ensure no non-linearity effects are present).

7.5 Complex Depth Maps

7.5.1 Introduction

Three real scenes with a variety of objects, colours and textures were imaged with a colour camera. Eight images were taken for a given f-number and then averaged to reduce the additive noise component. The images were processed using the implementation of Ens and Lawrence's DFD algorithm with real PSF data assuming a Gaussian model that was collected using the knife-edge based technique incorporating the non-uniform illumination model discussed in Chapters 3 and 4. The statistical-based normalisation algorithm was employed as it performed better than the alternatives summarised in Section 7.3.3.

The images captured with the colour camera were converted to monochrome using an equal weighting of the colour planes, i.e. $(\alpha, \beta, \gamma) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$. The colour pre-processing algorithms, such as PCA, were not employed because they were adversely affected by the multiplicative noise present in the camera, as shown in the previous chapter. The more focused image (i.e. that taken with f/5.6) was textured mapped on to the depth map using MATLAB to produce a $2\frac{1}{2}$ D image. The texture map was employed because it aids the viewer in locating the objects in a scene.

7.5.2 Test 1: Wooden Man with Plastic Football

An artists' wooden man was set up to hold a smooth plastic ball and placed in front of a texture, that was printed on a colour laser printer, as shown in Figure 7.12. The individual pentagons on the plastic-coated ball were essentially textureless and so black dots were added to aid the DFD algorithm. The wooden man possessed sufficient natural texture due to the woodgrain. A high resolution image of a section of red stone (stone_09) provided a suitable backdrop; during experimentation it was found that the small dynamic range of the camera restricted the texture that could be used.

Figure 7.12: Images of a wooden figure with ball using f/5.6 (left) and f/2.8 (right)

The texture-mapped depth map produced using the DFD algorithm is shown in Figure 7.13. For the purposes of analysing the map, small regions (shown labelled in Figure 7.14) that could be considered to be at approximately constant depths were used. The mean depth of the region was then compared to the actual measurements made with a ruler and the results are shown in Table 7.5.

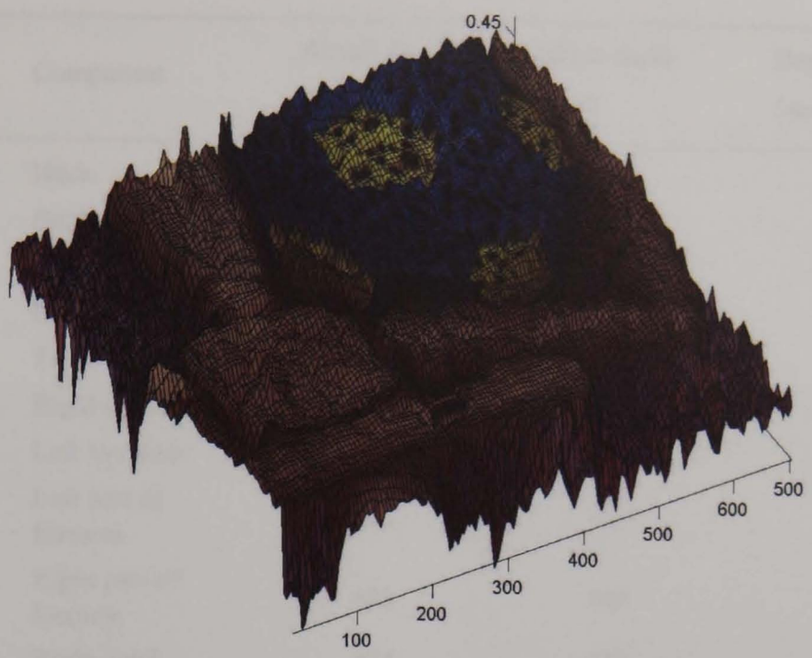


Figure 7.13: Texture mapped depth map of wooden figure with a ball

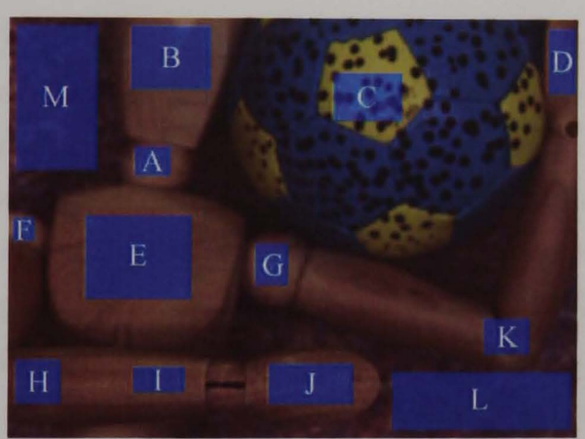


Figure 7.14: Labelled image of a wooden man with a plastic football

The man's right arm and hand (labelled H, I and J) are the closest parts to the camera and they have been resolved well and the resulting depth map is quite smooth. The depth map of the ball is particularly noisy, and this was attributed to the texture having strong blue components, which are known to be more noisy than red and green textures, as shown in Section 7.4.

Table 7.5. Analysis of the regions of Test 1

Label	Component	Actual depth / mm	Mean depth / mm	Depth error / mm
A	Neck	486	474	-12
B	Head	483	469	-14
C	Front of football	462	455	-7
D	Left hand	476	453	-23
E	Torso	480	475	-5
F	Right shoulder	490	476	-14
G	Left shoulder	490	476	-14
H	Left part of forearm	440	429	-11
I	Right part of forearm	430	424	-6
J	Right hand	434	423	-11
K	Elbow joint	480	469	-11
L	Background (right)	535	534	-1
M	Background (left)	535	538	3

The results show that the depth has been almost always under-estimated and this is consistent with the theory presented in Section 5.2.2 concerning the presence of noise in image 2. When the depth is under-estimated, the optimum convolution ratio is too small in the spatial domain, thus suggesting that there was too little change in the defocus between the images. Defocusing acts as a low-pass filter and thus the amplitude of the high frequency components in the defocused image are reduced, however, noise is not blurred and thus it becomes more apparent at high frequencies, thus increasing the spread of the convolution ratio and hence under-estimating the depth.

The best depth results were produced by the background regions that were perpendicular to the optical axis. The worse depth results were produced by the left hand (label D), which was curved. Ens and Lawrence's algorithm was based on the assumption that the depth is constant within a region and thus violations of the this assumption can be expected to produce significant depth errors.

It is known from experiments in Section 6.5.2 that the depth error increases with distance from the camera. The head (label B) and the torso (label E) of the wooden man both have the same texture and are at approximately the same depth, but the mean depth error is nearly three times greater for the head and this was believed to be due to its curvature, whereas the torso has a smoother change with depth.

7.5.3 Test 2: Wooden Man Holding Chess Piece

The randomly coloured checkerboard pattern performed well in experiments in the previous chapter and this was due to the good textural content. It was used as a backdrop for the wooden man, but this time with a wooden chess piece. The defocused images used taken with apertures of f/5.6 and f/2.8 are shown in Figure 7.15. The resulting depth map is shown in Figure 7.16.

Figure 7.15: Images of a wooden figure with a chess piece using f/5.6 (left) and f/2.8 (right)

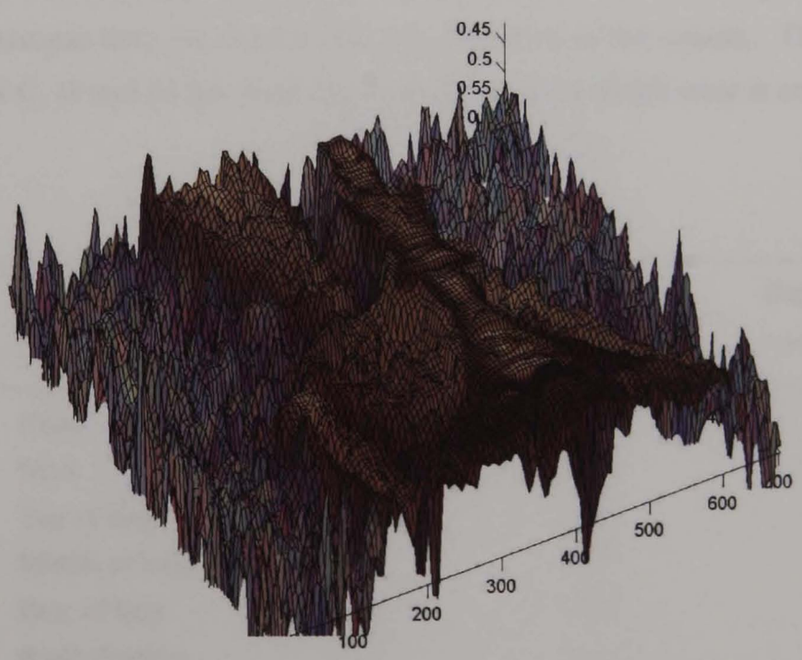


Figure 7.16: Texture mapped depth map of a wooden figure with a chess piece

Figure 7.17: Labelled image of a wooden man holding a chess piece

The specific regions considered are shown labelled in Figure 7.17 and the results compared to the actual measurements in Table 7.6. The depth of the background of the scene, although perpendicular to the optical axis, has been resolved badly. It is at 0.581m from the camera and is therefore outside the range of 0.41m to 0.52m originally proposed, however, it was used for testing purposes. The large distance from the camera has resulted in very defocused regions, making it very sensitive to noise in the system.

The hands of the wooden man (labelled H and I) produced very small depth errors and this is probably because they are close to the focus position of the camera. The wooden chess king (labels C, D and E) has been resolved well and the depth error is consistent at about 16mm.

Table 7.6. Analysis of the regions of Test 2

Label	Component	Actual depth / mm	Mean depth / mm	Depth error / mm
A	Head	517	496	-21
B	Neck	523	506	-17
C	Top of king	450	434	-16
D	Middle of king	445	429	-16
E	Base of king	440	423	-17
F	Right shoulder joint	535	511	-14
G	Torso	520	516	-4
H	Right hand	438	437	-1
I	Left hand	438	438	0
J	Left shoulder joint	527	506	-10
K	Background (left)	581	602	21
L	Background (right)	581	610	29

7.5.4 Test 3: Toy Dog

A toy dog with a complex depth map due to its construction was imaged to provide a difficult test. The defocused images used are shown in Figure 7.18. The wool gives a very good texture that is 3D, i.e. the texture is not purely in intensity, as with the wooden pieces of the previous two images. The eyes are shiny plastic and are essentially texture-less, so it could not be expected that the depth would be found accurately.

Figure 7.18: Images of a toy dog with ball using f/5.6 (left) and f/2.8 (right)

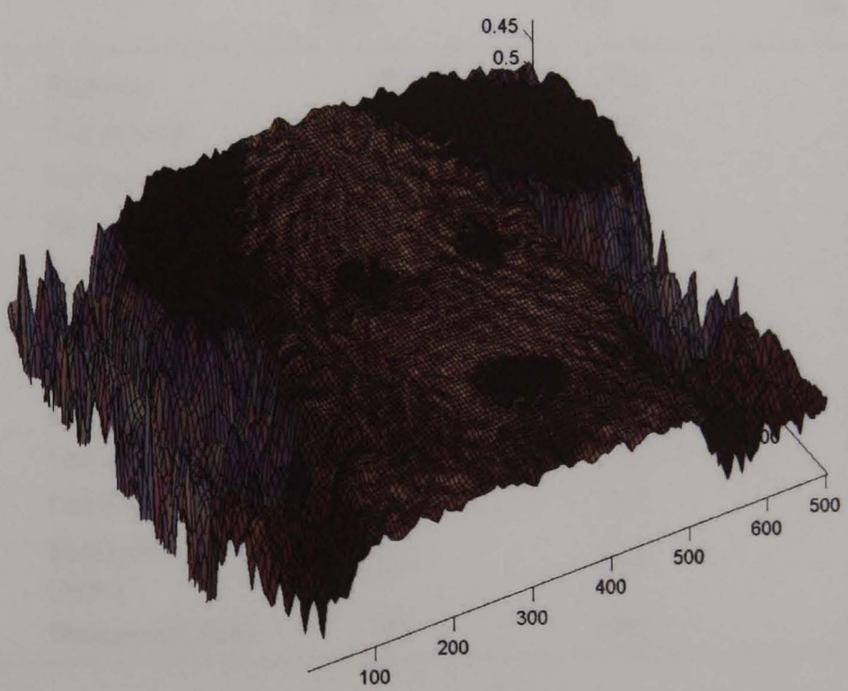


Figure 7.19: Texture mapped depth map of the toy dog

Figure 7.20: Labelled image of a toy dog

The woollen construction of the toy dog ensured plenty of texture. The eyes were essentially textureless, but median filtering the depth map has reduced the depth error there and in fact the depth error is consistent for both eyes.

Table 7.7. Analysis of the regions of Test 3

Label	Component	Actual depth / mm	Mean depth / mm	Depth error / mm
A	Right ear	460	452	-8
B	Top of head	438	428	-10
C	Left ear	460	446	-14
D	Back right leg	530	521	-9
E	Front right leg	460	452	-8
F	Nose	428	418	-10
G	Right eye	440	428	-12
H	Left eye	440	428	-12
I	Left front leg	475	446	-29
J	Left back leg	530	522	-8
K	Background (right)	580	588	8
L	Background (left)	580	605	25

7.5.5 Conclusion

The correlation coefficient (defined in Section 4.2.3) of the actual depth and the depth error was calculated based on the data for each test. The depth error is plotted as a function of the actual depth in Figure 7.21.

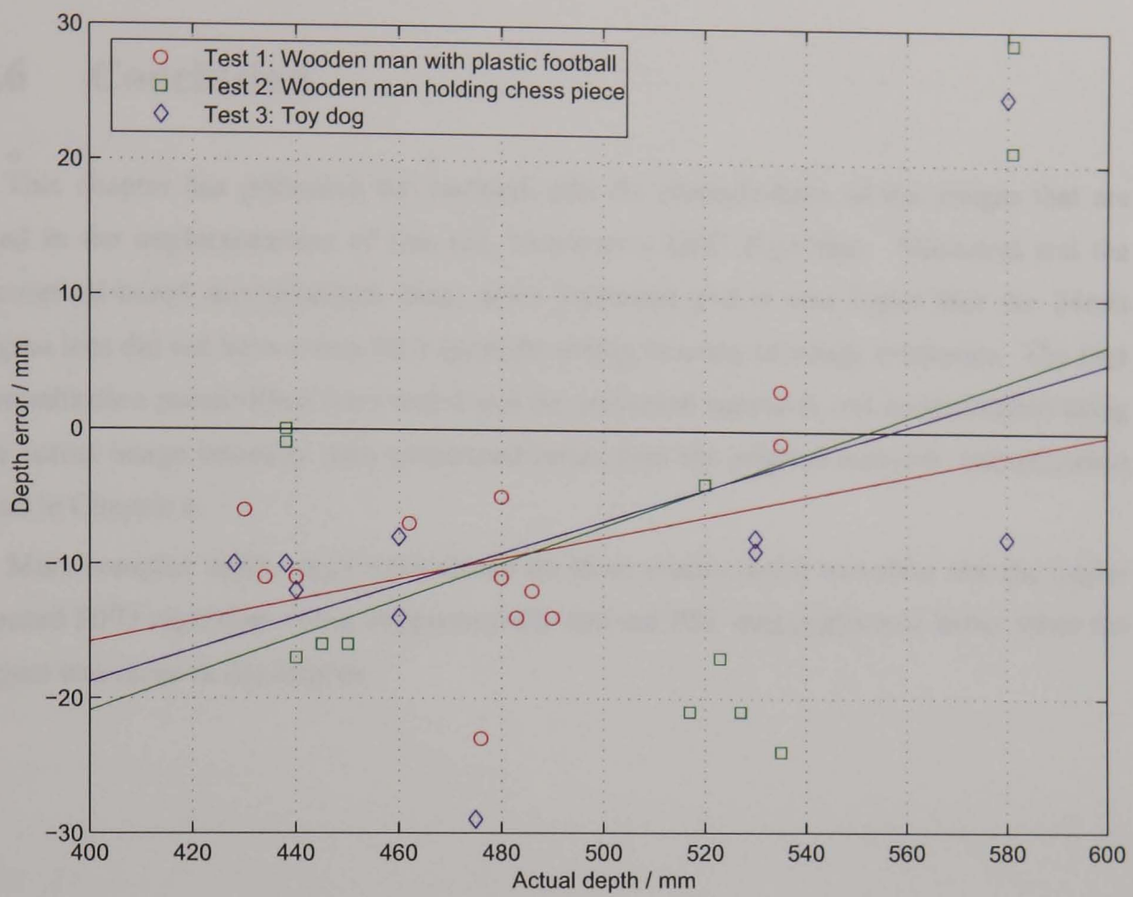


Figure 7.21: Mean depth error plotted as a function of the actual depth for the three tests

The depth range employed in the experiments and the correlation coefficient are summarised in Table 7.8. The positive correlation coefficient shows that as the depth of a point is increased, the mean error increases, but as it is generally negative, it makes the mean depth error better. When the depth is under-estimated, image 2 is too noisy and when the depth is over-estimated, image 1 is too noisy. From Figure 7.21 it can be seen that the depth is under-estimated up to a depth of around 550mm for Tests 2 and 3 and then over-estimated at larger depths. The reason for this would require further analysis and it would have to centre on the effect of multiplicative noise as a function of the relative defocusing between the images.

Table 7.8. Summary of the complex scenes

Test	Correlation coefficient	Depth range / mm	Mean depth error / mm
1	0.38	430 - 535 (105)	-9.7
2	0.44	440 - 581 (141)	-7.3
3	0.55	428 - 580 (152)	-8.6

The depth has been under-estimated in all three scenes, which is consistent with the theory of Section 5.2.2.

7.6 Conclusion

This chapter has presented the research into the normalisation of the images that are used in the implementation of Ens and Lawrence's DFD algorithm. Statistical and the theoretical-based normalisation ideas were presented and it was found that the 24mm Sigma lens did not have a true f/2.8 aperture setting in terms of image irradiance. The four normalisation possibilities were tested and the statistical approach and normalisation using the actual image intensity ratio performed better than the original min-max normalisation used in Chapter 6.

More complex depth maps were shown for three scenes and it was clear that the implemented DFD algorithm using experimentally derived PSF data performed better when the object was close to the camera.

Chapter 8

Conclusions and Future Work

8.1 Introduction

The research was divided into two sections, namely the measurement of the PSF of a defocused imaging system and the development of a colour image pre-processing stage for depth-from-defocus. In this chapter the conclusions of the research are drawn and then further work that could be done is outlined.

In summary, the main contributions of this thesis concerning the measurement of the PSF of a defocused imaging system are:

- Generalised Gaussian model of the PSF (14 times better than the pillbox, 8 times better than the Gaussian and 5 times better than the sum of Fermi-Dirac functions);
- Non-uniform illumination model of the lightbox (reduced the MSE by 25%).

The main theoretical contributions in the field of DFD are:

- More accurate depth maps can be found using colour images instead of monochrome images;
- In the presence of additive noise, PCA produces better depth maps than monochrome, however, with multiplicative noise it performs worse;
- SNR can be boosted through colour mixing assuming an additive noise model, and thus producing better depth maps;
- Maximising the fractal dimension through colour mixing, where a least-squares fit assuming fractional Brownian motion was used to measure the FD, produced worse depth accuracy than monochrome;
- A projected colour pattern and the LCM algorithm have shown that better depth localisation is possible.

The key theoretical contribution of this work is that although DFD is a frequency domain approach, better sampling and analysis of the spectral domain (i.e. using a colour camera) holds potential for more accurate depth maps than have been produced before using existing monochrome algorithms.

8.2 Point Spread Function Measurement

8.2.1 Introduction

The Point Spread Function (PSF) is very important for characterising optical systems and Chapters 3 and 4 discussed methods of finding the PSF. The research focused on using a knife-edge based technique originally developed by Reichenbach *et al.* [55] and improved by Tzannes and Mooney [56] and converted to 2D by Staunton [57].

8.2.2 Analysis of Research and Original Contribution

Staunton [57] used a lightbox with a knife-edge to produce a sharp intensity transition and assumed that the step edge had uniform brightnesses in the upper and lower regions, which was a reasonable approximation as his work only considered a focused camera system. This research considered a camera system that could be focused or defocused and due to the increased spatial extent of the ESF and the construction of the lightbox, a non-uniform illumination model was created. The MSE between the fitted ESF and a model of the ESF decreased by 25% when the non-uniform illumination was incorporated, which is clearly a significant decrease.

Space-invariance was assumed and so the maximum available number of ESF profiles in the image were used to reduce noise and produce a super-resolution ESF. The knife edge was rotated in 10 degree increments about the centre of the image captured by the camera to build up the 2D PSF.

The biggest problem was to process the measured ESF to obtain the PSF. Various PSF models were considered: geometrical optics model (the pillbox); sum of Fermi-Dirac functions (as proposed by Tzannes and Mooney [56]); Gaussian; and the Generalised Gaussian (the novel model proposed). The noise level of the camera was sufficient to preclude the use of a five-point forward-difference formula that performs differentiation. Thus, a more advanced approach was sought. Chartrand's regularised numerical differentiation algorithm [126] is not based on any model and thus has more flexibility than assuming a given PSF model. The regularised numerical differentiation could not adequately account for the non-uniform illumination and further the results were poor when compared to the Generalised Gaussian.

The proposed model of a defocused camera system, the Generalised Gaussian, was 14 times better than the pillbox and 8 times better than the Gaussian model with the 24mm

Sigma photographic lens. The sum of the Fermi-Dirac functions was 5 times worse than the Generalised Gaussian, and thus had a better MSE than the Gaussian and pillbox models, however, the non-uniform illumination could not be taken into account. Further, the shape of the 1D PSF was often asymmetric, which was not expected in a well-corrected lens.

The camera movement was automated using an x-stage and controlled through the parallel port of a computer with software written in Visual Basic. The combination of hardware and the MATLAB software implementation produced results for the 16mm video lens and the 24mm Sigma lens; and the former was diagnosed to suffer from spherical aberration as well as possibly coma and astigmatism. The 24mm Sigma lens had a PSF that was circularly symmetric to a good approximation, thus suggesting that the aberrations, if present at all, were negligible.

8.2.3 Future Work

The output of Chartrand's numerical differentiation algorithm [126] was highly dependent on the choice of the regularisation parameter. More analysis into the parameter may help to alleviate some of the problems with the overall shape, but the fact that it cannot directly account for the non-uniform illumination is a hindrance. If a new lightbox was constructed that does not have a significant illumination change in either region then the regularised numerical differentiation, coupled with a better choice of the regularisation parameter, might be optimum.

The results in Section 4.7.4 showed that the fitted Generalised Gaussian had a standard deviation that was a smooth function of the depth of the lightbox, but the power was quite noisy. It was believed that the power as a function of depth should be smooth too. MATLAB's function least squares curve fitting routine (*lsqcurvefit*) was used, but different fitting algorithms could be investigated that fit the actual ESF to the model.

By assuming that the PSF was space-invariant, it was possible to use many ESF profiles to create a super-resolution ESF. In order to test this assumption, many smaller knife-edges spread throughout the image could be employed. For example, a matrix of 3×3 knife edges could be used to give nine super-resolution ESFs that could then be processed separately. By assuming a Gaussian model, for example, the change in the standard deviation as a function of the position in the image gives an indication as to its space-variant nature.

The colour images captured by the camera fitted with a Bayer filter were converted to monochrome using a very simple algorithm that reduced the spatial dimensions of the resultant image by half compared to that captured by the camera. Different demosaicing

algorithms could be analysed to determine their effect on the PSF. The 24mm Sigma lens was assumed to be an achromat, i.e. possess very little chromatic aberration, as it was a high quality photographic lens. However, by testing each colour plane separately, or even better using various sources with a restricted spectral band, the PSF for a given range of wavelengths could be determined. If one light source is used then better illumination for the lightbox, such as a fluorescent or xeon tube (with colour temperatures around 5000K), should be employed to give a more even spectral response. With the incandescent bulbs used in the experiments, the peak in the visible wavelengths is at the red end of the spectrum (with a colour temperature around 2500K), and thus it does not adequately allow the PSF to be determined as an average for all visible wavelengths.

Motorising the rotation of the lightbox and controlling it with the computer would be the last required hardware adjustment to make for a fully automated system for testing cameras and their associated lenses.

8.3 Colour Depth-From-Defocus

8.3.1 Introduction

Dimension reduction of colour images to an optimum monochrome image was shown using a Genetic Algorithm that finds the colour plane weighting given the known depth. The requirement of a known object depth makes the approach unusable in practice and deterministic methods were employed in an attempt to approximate the function performed by the GA using Principal Component Analysis, maximisation of the SNR, maximisation of the fractal dimension and LCM.

8.3.2 Analysis of Research and Original Contribution

As far as the author is aware, this is the first work done on colour DFD that uses two defocused RGB images. Hiura and Matsuyama [104] used a 3-colour camera to capture three images where each image plane was imaged with lenses of differing focal lengths. Murata and Kawamura [105] used a similar approach for Particle Image Velocimetry, but with two colour planes only.

The work began with the realisation that a monochrome camera can lose important textural information that is chromatic in nature. A GA was written to discover if there were optimum linear combinations of the colour planes and the result was affirmative. The GA could yield limited information about how it was achieving such good depth maps

and further it was capable of manipulating the noise present in an image to meet its desired goal.

In the presence of uncorrelated, additive noise PCA was found to be superior to using an equal weighting of the colour planes. An image corrupted by AWGN was scaled by the eigenvector with the largest eigenvalue and improvements of between 1.3 and 1.5 times were found over the monochrome case. However, multiplicative noise adversely affected its ability to produce eigenvectors that give a good SNR. A weighted PCA algorithm based on the noise variances of the colour planes, denoted NVA-PCA, was not sufficient to alleviate the problem in the experiments.

An algorithm was devised to maximise the SNR assuming an additive model and simulations showed that with noise that has the same variance (i.e. the noise is isotropic) in each colour plane that maximising the SNR and PCA produced essentially the same results. Evolving the solution using a GA is slower than using PCA and so the matrix-based solution should be used for efficiency. When the noise is non-isotropic, PCA is no longer optimum and the algorithm to maximise the SNR gave SNR improvements of around 2dB compared to PCA and 3dB compared to monochrome. The small increase in the SNR resulted in depth maps with a MSE that was between 3.4 and 7.8 times better than monochrome and 1.7 to 1.9 times better than PCA.

The algorithm to find (α, β, γ) to maximise the fractal dimension gave worse depth maps than using both the monochrome and PCA approaches. This was traced to the reduction in the SNR by maximising the FD and so to be usable the SNR could be taken into account through a multi-objective optimisation approach.

The *Localisation through Colour Mixing* (LCM) algorithm was specifically designed to reduce the windowing and image overlap problem. The scaling constants (α, β, γ) were derived using the Moore-Penrose matrix inverse to give the best approximation to a monochrome image with an impulse at the centre pixel. The approach required a random colour checkerboard pattern to be projected onto the scene using a telecentric projector to ensure the pixels have the same size on the camera regardless of depth. Due to the lack of this equipment, only simulations could be performed. LCM was found to give depth maps that were between 7.3 and 9.4 times better than monochrome and 1.7 and 2.2 times better than PCA. The SNR was reduced by improving the localisation, thus showing a trade-off that must be carefully managed in practice.

8.3.3 Future Work

First and foremost, the lack of an adequate noise model for the camera hindered the construction of the algorithms and the underlying assumption that the noise was additive was clearly incorrect. A de-noising pre-process would be very useful before the resulting denoised images were applied to the PCA, maximisation of the FD or LCM algorithms. Alternatively, algorithms such as PCA, maximisation of the SNR and LCM need to be reformulated to be robust in the presence of multiplicative noise.

An Artificial Neural Network (ANN) is composed of simple processing elements called neurons that can be used to model processes through a training procedure. An ANN could be employed to take the defocused colour images and return the optimum weights, denoted (α, β, γ). The GA or the best point on a response surface [189] [190] [191] of (α, β, γ) could be used as the training input. The problem with presenting even a single image to the ANN is that for a window size of 32×32 with three colour planes, the number of inputs would be 3072. In order to reduce the number of inputs the statistics of each colour plane could be entered, including the mean, variance, skewness and kurtosis of each colour plane along with covariances between planes.

The dynamic range of a CCD camera is very much less than the human visual system and it was a difficult to ensure that parts of the scene were not saturated while other parts were too dark to be imaged for subsequent tests. A much larger dynamic range could be achieved by using multiple exposure times and then reconstructing the scene.

The research into colour DFD used a linear approach to creating a monochrome image and non-linear approaches could be investigated. A more advanced idea would be to employ an explicit multi-channel approach (instead of an implicit approach investigated in this thesis) that can take into account the correlations between the colour planes and the wavelength dependent nature of the PSF.

Appendix A

Derivation of the Edge Spread Functions

A.1 Introduction

The Point Spread Function (PSF) characterises an optical system and it is important to know the PSF accurately for Depth from Defocus (DFD) work for precise recovery of the depth of objects in a scene. The PSF can be measured by imaging a step edge in intensity to find the Edge Spread Function (ESF) and then differentiating the response as shown by [55] [56] [57]. Numerical differentiation of discrete data is problematic when noise is present, but it is possible to assume the PSF comes from a particular family of shapes. The experimentally obtained ESF can be fitted (in a least squares sense) to a model ESF formed from a defocus blurred ideal step with a particular PSF shape and from the fit the PSF parameters can be determined. In this Appendix the general ESFs are derived for steps that account for the experimental issues of non-uniform illumination when the PSF is a mathematically defined pillbox, Gaussian and Generalised Gaussian. The non-uniform illumination is modelled as a linear change in intensity with distance as this fitted with the experimental results.

Previously the two PSF models most commonly employed in depth-from-defocus are the pillbox and Gaussian models, but the Generalised Gaussian was shown to be a good contender. The main problem with the Generalised Gaussian function is that it strongly resists being manipulated mathematically by virtue of its non-integer power.

A.2 Edge Spread Function Model

Consider a one-dimensional step edge where the brightness of the upper and lower levels have a linear dependence on position. The intensity of the bright region is given by

$$y_1(x) = m_1 x + c_1 \tag{A1}$$

and the intensity of the dark region is given by

$$y_2(x) = m_2 x + c_2 \tag{A2}$$

where x is the position measured in pixels, m_i is the gradient of the brightness and c_i is the brightness at the discontinuity where $i \in [1, 2]$ as shown in Figure A.1. The discontinuity occurs at $x = x_0$, thus the piecewise function representing the ideal (non-blurred) step is given by

$$s(x) = \begin{cases} m_1 x + c_1 & x \le x_0 \\ m_2 x + c_2 & x > x_0 \end{cases} \tag{A3}$$

The unit step function is defined as

$$u(x) = \begin{cases} 0 & x < 0 \\ 1 & x \ge 0 \end{cases} \tag{A4}$$

and thus the ideal step with non-uniform illumination $s(x)$ can be written as

$$s(x) = [m_1 x + c_1]\, u(x_0 + x) + [m_2 x + c_2]\, u(x - x_0). \tag{A5}$$

Figure A.1 below shows an example of a step with non-uniform illumination where $m_1 = 2$, $m_2 = -2$, $c_1 = 250$, $c_2 = 50$ and $x_0 = 2$. The parameters of the non-uniform illumination were chosen to exaggerate the actual effect found in experimental work to make the resulting ESF easier to see visually.

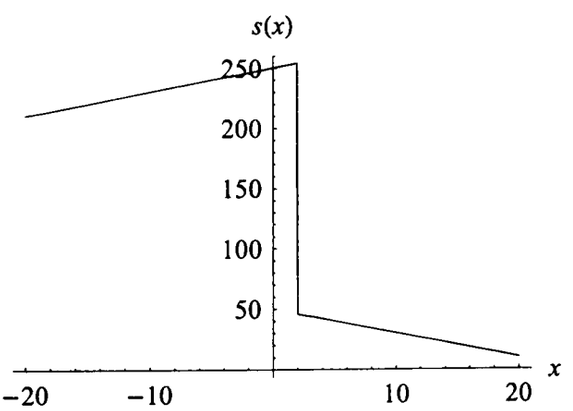


Figure A.1: Step with non-uniform illumination

The ESF $f(x)$ is given by the convolution of the ideal step $s(x)$ with the PSF $h(x)$ and it can be denoted

$$f(x) = s(x) * h(x) \tag{A6}$$

where $*$ denotes linear convolution. The convolution integral allows the equation to be re-written as

$$f(x) = \int_{-\infty}^{\infty} s(\xi)\, h(x - \xi)\, d\xi = \int_{-\infty}^{\infty} s(x - \xi)\, h(\xi)\, d\xi \tag{A7}$$

and substituting in A5 gives the general ESF as

$$f(x) = \int_{-\infty}^{\infty} h(\xi)[m_1(x-\xi)+c_1]\,u(x_0-x+\xi)\,d\xi + \int_{-\infty}^{\infty} h(\xi)[m_2(x-\xi)+c_2]\,u(x-\xi-x_0)\,d\xi \tag{A8}$$

The shifted unit steps in A8 mean that the limits of the integration can be reduced as

$$u(x_0-x+\xi) = \begin{cases} 0 & \xi < x-x_0 \\ 1 & \xi \geq x-x_0 \end{cases} \tag{A9}$$

and

$$u(x-\xi-x_0) = \begin{cases} 0 & \xi > x-x_0 \\ 1 & \xi \leq x-x_0 \end{cases} \tag{A10}$$

and thus the ESF becomes

$$f(x) = \int_{x-x_0}^{\infty} h(\xi)[m_1(x-\xi)+c_1]\,d\xi + \int_{-\infty}^{x-x_0} h(\xi)[m_2(x-\xi)+c_2]\,d\xi \tag{A11}$$

For the purposes of the derivation it is useful to split up the ESF into two halves so that

$$f(x) = \Lambda_b(x) + \Lambda_d(x) \tag{A12}$$

where $\Lambda_b(x)$ corresponds to contribution due to the bright region and it is given by

$$\Lambda_b(x) = \int_{x-x_0}^{\infty} h(\xi)[m_1(x-\xi)+c_1]\,d\xi \tag{A13}$$

and $\Lambda_d(x)$ corresponds to the dark region where

$$\Lambda_d(x) = \int_{-\infty}^{x-x_0} h(\xi)[m_2(x-\xi)+c_2]\,d\xi. \tag{A14}$$

A.3 Pillbox PSF Model

Now consider a pillbox PSF with unit area that is given by

$$h_p(x) = \frac{1}{2\sigma}[u(x+\sigma)-u(x-\sigma)] \tag{A15}$$

where σ is the radius of the pillbox (and hence the blur circle). Figure A.2 shows a pillbox PSF where the radius $\sigma = 5$.

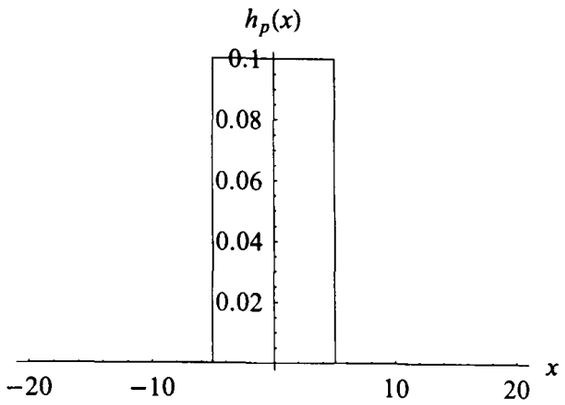


Figure A.2: The pillbox PSF with a radius $\sigma = 5$

The ESF due to the dark region is given by (A13) and substituting (A15) in gives

$$\Lambda_b(x) = \frac{1}{2\sigma} \int_{x-x_0}^{\infty} [u(x+\sigma) - u(x-\sigma)][m_1(x-\xi) + c_1]\, d\xi. \tag{A16}$$

The piecewise nature of the PSF means that the ESF needs to be computed piecewise too. If $x - x_0 < -\sigma$ then

$$\Lambda_b(x) = \tfrac{1}{2\sigma} \int_{-\sigma}^{\sigma} [m_1(x-\xi) + c_1]\, d\xi \tag{A17}$$

$$= m_1 x + c_1 \tag{A18}$$

and if $-\sigma \le x - x_0 \le \sigma$ then

$$\Lambda_b(x) = \tfrac{1}{2\sigma} \int_{x-x_0}^{\sigma} [m_1(x-\xi) + c_1]\, d\xi \tag{A19}$$

$$= -\tfrac{1}{4\sigma} \left((2c_1 + m_1(x + x_0 - \sigma))(x - x_0 - \sigma)\right) \tag{A20}$$

and finally if $\sigma < x - x_0$ then $\Lambda_b(x) = 0$.

A similar analysis for the dark region gives

$$\Lambda_d(x) = \frac{1}{2\sigma} \int_{-\infty}^{\infty} [u(x+\sigma) - u(x-\sigma)][m_2(x-\xi) + c_2]\, d\xi \tag{A21}$$

and again the equation must be considered piecewise. If $x - x_0 < -\sigma$ then $\Lambda_d(x) = 0$ and if $-\sigma \le x - x_0 \le \sigma$ then

$$\Lambda_d(x) = \tfrac{1}{2\sigma} \int_{-\infty}^{\infty} [m_2(x-\xi) + c_2]\, d\xi \tag{A22}$$

$$= \tfrac{(x-x_0+\sigma)(2c_2 + m_2(x+x_0+\sigma))}{4\sigma} \tag{A23}$$

and $\sigma < x - x_0$ then

$$\Lambda_d(x) = m_2 x + c_2 \tag{A24}$$

Combining the results gives the ESF for a pillbox PSF where the step has non-uniform illumination as

$$f_p(x) = \begin{cases} m_1 x + c_1 & x - x_0 < -\sigma \\ \frac{1}{4\sigma}[-(2c_1 + m_1(x + x_0 - \sigma))(x - x_0 - \sigma) + & -\sigma \le x - x_0 \le \sigma \\ \quad (x - x_0 + \sigma)(2c_2 + m_2(x + x_0 + \sigma))] & \\ m_2 x + c_2 & \sigma < x - x_0 \end{cases} \tag{A25}$$

and an example of the shape of the ESF is shown in Figure A.3 where $\sigma = 5$ and the step with non-uniform illumination has the same shape as used in Figure A.1. The original (focused) step has been shown as a dashed line for comparison with the defocus blurred step, shown with the solid line.

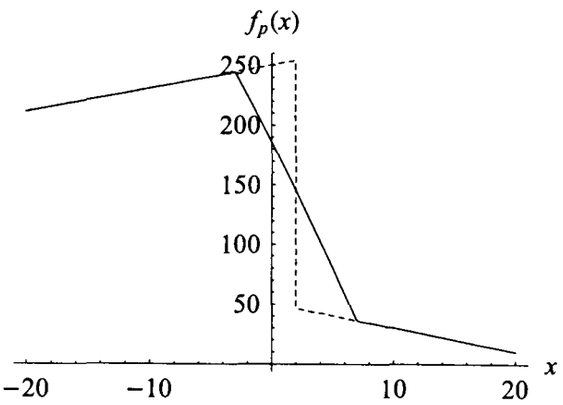


Figure A.3: ESF with a pillbox PSF where $\sigma = 5$

A.4 Gaussian PSF Model

Now consider a Gaussian PSF with unit area that is given by

$$h_g(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\frac{(x - \hat{x})^2}{\sigma^2}\right\} \tag{A26}$$

where σ is the standard deviation and it is assumed that the mean $\hat{x}$ is zero. Figure A.4 shows the Gaussian with $\sigma = 5$.

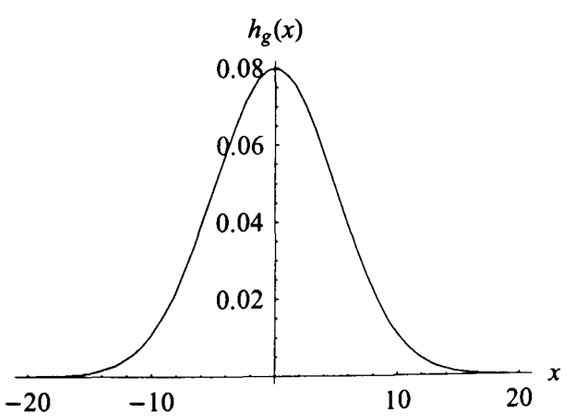


Figure A.4: Gaussian PSF where $\sigma = 5$

Substituting the Gaussian PSF A26 into (A11) gives the ESF as

$$f_g(x) =$$

$$\frac{1}{\sigma\sqrt{2\pi}}\int_{x-x_0}^{\infty} e^{-\frac{1}{2}\frac{\xi^2}{\sigma^2}}[m_1(x-\xi)+c_1]\,d\xi + \frac{1}{\sigma\sqrt{2\pi}}\int_{-\infty}^{x-x_0} e^{-\frac{1}{2}\frac{\xi^2}{\sigma^2}}[m_2(x-\xi)+c_2]\,d\xi \tag{A27}$$

Due to the discontinuity in the brightness the ESF can be considered to be composed of two distinct regions – bright and dark – as before. The parts of the ESF corresponding to the bright $\Lambda_b(x)$ and dark regions $\Lambda_d(x)$ are

$$\Lambda_b(x) = \frac{1}{\sigma\sqrt{2\pi}}\int_{x-x_0}^{\infty} e^{-\frac{1}{2}\frac{\xi^2}{\sigma^2}}[m_1(x-\xi)+c_1]\,d\xi \tag{A28}$$

and

$$\Lambda_d(x) = \frac{1}{\sigma\sqrt{2\pi}}\int_{-\infty}^{x-x_0} e^{-\frac{1}{2}\frac{\xi^2}{\sigma^2}}[m_2(x-\xi)+c_2]\,d\xi \tag{A29}$$

It is necessary to perform the integration by parts and it can be shown using Mathematica that

$$\Lambda_b(x) = \frac{1}{2}\left(-m_1\,\sigma\sqrt{\frac{2}{\pi}}\,e^{-\frac{1}{2}\frac{(x-x_0)^2}{\sigma^2}} + (m_1\,x+c_1)\left(1-\operatorname{erf}\left(\frac{x-x_0}{\sigma\sqrt{2}}\right)\right)\right) \tag{A30}$$

and

$$\Lambda_d(x) = \frac{1}{2}\left(-m_2\,\sigma\sqrt{\frac{2}{\pi}}\,e^{-\frac{1}{2}\frac{(x-x_0)^2}{\sigma^2}} + (m_2\,x+c_2)\left(1+\operatorname{erf}\left(\frac{x-x_0}{\sigma\sqrt{2}}\right)\right)\right) \tag{A31}$$

where the error function $\operatorname{erf}(x)$ is defined as

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt \tag{A32}$$

and a plot of the function is shown in Figure A.5.

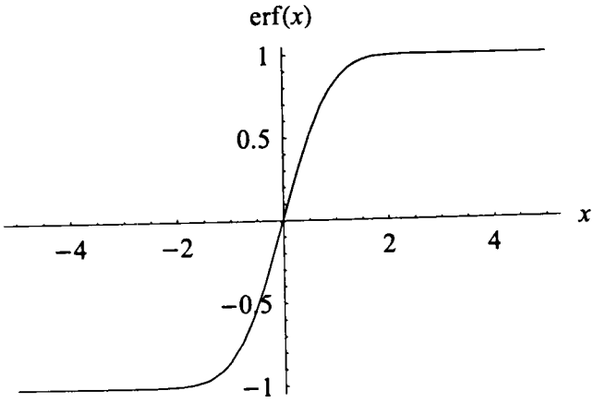


Figure A.5: The error function $\operatorname{erf}(x)$

The Gaussian defocused step taking into account non-uniform illumination is given by

$$f_g(x) = \frac{1}{2}\left[-(m_1 + m_2)\,\sigma\sqrt{\frac{2}{\pi}}\,e^{-\frac{1}{2}\frac{(x-x_0)^2}{\sigma^2}} + \right.$$
$$\left.(m_1 x + c_1)\left(1 - \mathrm{erf}\left(\frac{x - x_0}{\sigma\sqrt{2}}\right)\right) + (m_2 x + c_2)\left(1 + \mathrm{erf}\left(\frac{x - x_0}{\sigma\sqrt{2}}\right)\right)\right]. \tag{A33}$$

and Figure A.6 shows the shape of the ESF if a Gaussian with $\sigma = 5$ is used.

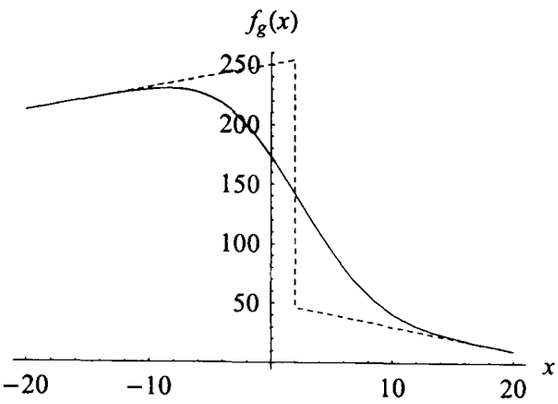


Figure A.6: ESF when the PSF is a Gaussian with $\sigma = 5$

A.5 Generalised Gaussian PSF Model

A new model proposed for the PSF is the Generalised Gaussian and it is given by

$$h_G(x) = \frac{p^{1-\frac{1}{p}}}{2\,\sigma\,\Gamma\left(\frac{1}{p}\right)} \exp\left\{-\frac{1}{p}\,\frac{|x - \hat{x}|^p}{\sigma^p}\right\} \tag{A34}$$

where $\Gamma(\cdot)$ is the Gamma function, σ is the standard deviation of the function, $\hat{x}$ is the mean, p is the power and $|\cdot|$ represents the modulus. It is assumed that the mean $\hat{x}$ is zero, thus this is no phase shift. When $p = 2$ the Generalised Gaussian reduces to a normal Gaussian. Figure A.7 below shows Generalised Gaussians for ($p = 1$, $\sigma = 5$) and ($p = 4$, $\sigma = 5$)

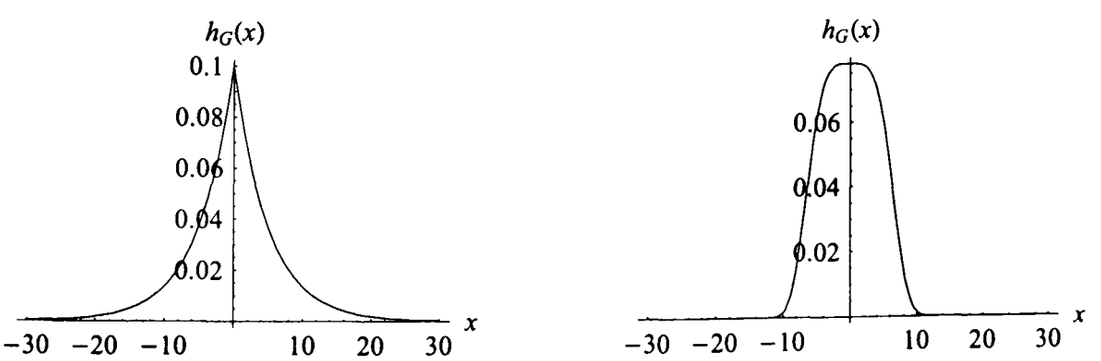


Figure A.7: Generalised Gaussian PSFs where (left) $p = 1$ and $\sigma = 5$; (right) $p = 4$ and $\sigma = 5$

From the figure it can be seen that the power p controls the pointedness of the peak and σ specifies the spread, as with a normal Gaussian function. Using the general form of the ESF given in (A11) and the Generalised Gaussian PSF (A34) results in

$$f_G(x) = \frac{p^{1-\frac{1}{p}}}{2\sigma\Gamma\left(\frac{1}{p}\right)} \int_{x-x_0}^{\infty} \exp\left\{-\frac{1}{p}\frac{|\xi|^p}{\sigma^p}\right\}[m_1(x-\xi)+c_1]\,d\xi + \frac{p^{1-\frac{1}{p}}}{2\sigma\Gamma\left(\frac{1}{p}\right)} \int_{-\infty}^{x-x_0} \exp\left\{-\frac{1}{p}\frac{|\xi|^p}{\sigma^p}\right\}[m_2(x-\xi)+c_2]\,d\xi \tag{A35}$$

Mathematica and Maple were employed in an attempt to simplify the equations but to no avail and so in order to calculate the ESF assuming a Generalised Gaussian numerical integration was employed. Figure A.8 below show the ESFs for the Generalised Gaussians described above.

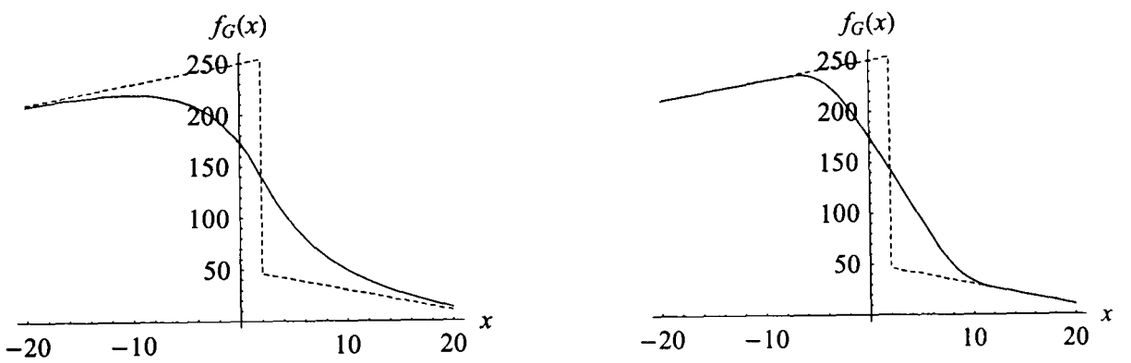


Figure A.8: The ideal steps (dashed lines) and ESFs (solid lines) assuming Generalised Gaussian PSFs with (left) $p = 1$ and $\sigma = 5$; (right) $p = 4$ and $\sigma = 5$.

Appendix B

Analysis of Linear Transformations of Images for Colour Depth-from-Defocus

B.1 Introduction

Ens and Lawrence's [58] [59] DFD algorithm requires two differently defocused images to be employed to determine the depth. The colour images are pre-processed to give a monochrome image $M(x, y)$ using

$$M_1(x, y) = \alpha_1 R_1(x, y) + \beta_1 G_1(x, y) + \gamma_1 B_1(x, y) \tag{B1}$$

$$M_2(x, y) = \alpha_2 R_2(x, y) + \beta_2 G_2(x, y) + \gamma_2 B_2(x, y) \tag{B2}$$

where $R_1(x, y)$, $G_1(x, y)$ and $B_1(x, y)$ are the red, green and blue components respectively of image 1 and $(\alpha_1, \beta_1, \gamma_1)$ are the real scaling constants. In the general case for two images there will be two sets of scaling constants, $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$ and considerations as to what restrictions must be placed on the constants so that DFD can still be performed accurately are important and necessary.

B.2 Mathematical Analysis of Monochrome Case

In order to solve the problem, linear transformations of the two monochrome images are considered in this section and then the specific problem is examined in the next section.

Consider two defocused images $i_1(x, y)$ and $i_2(x, y)$ where

$$i_1(x, y) = f(x, y) * h_1(x, y) \tag{B3}$$

and

$$i_2(x, y) = f(x, y) * h_2(x, y) \tag{B4}$$

where the image that would be formed with a pinhole camera is denoted $f(x, y)$ and $h_1(x, y)$ and $h_2(x, y)$ are the Point Spread Functions (PSFs), which are directly related to the camera parameters and the depth of the object.

Ens and Lawrence's DFD algorithm [58] [59] searches through the known set of pre-computed convolution ratios $h_3(x, y)$ to find the particular one that gives the lowest mean square error, i.e.

$$\min \sum_{x,y} (i_1(x, y) * h_3(x, y) - i_2(x, y))^2 \tag{B5}$$

where $*$ denotes linear convolution. In effect the algorithm searches for the best convolution ratio such that blurring the defocused image taken by camera 1 approximates that taken with camera 2.

The difference between the images without using the mean square measure is given by

$$d(x, y) = i_1(x, y) * h_3(x, y) - i_2(x, y) \tag{B6}$$

and substituting (B3) and (B4) into (B6) gives

$$d(x, y) = [f(x, y) * h_1(x, y)] * h_3(x, y) - [f(x, y) * h_2(x, y)]. \tag{B7}$$

The associative and distributive laws of convolution mean that the difference $d(x, y)$ can be written as

$$d(x, y) = f(x, y) * [(h_1(x, y) * h_3(x, y)) - h_2(x, y)] \tag{B8}$$

and it can be clearly seen that, assuming no noise, using the correct $h_3(x, y)$ sets the term in square brackets is zero.

Consider now the effect when the ideal defocused images $i_1(x, y)$ and $i_2(x, y)$ have undergone a linear transformation to produce two images $i_1'(x, y)$ and $i_2'(x, y)$ given by

$$i_1'(x, y) = \phi_1 i_1(x, y) + \psi_1 \tag{B9}$$

and

$$i_2'(x, y) = \phi_2 i_2(x, y) + \psi_2. \tag{B10}$$

The difference $d(x, y)$ when the two images employed have undergone a linear transformation is given by substituting (B9) and (B10) into (B6) to give

$$d(x, y) = [\phi_1 i_1(x, y) + \psi_1] * h_3(x, y) - [\phi_2 i_2(x, y) + \psi_2]. \tag{B11}$$

The defocused images $i_1(x, y)$ and $i_2(x, y)$ are given by (B3) and (B4) and so

$$d(x, y) = [\phi_1 \{f(x, y) * h_1(x, y)\} + \psi_1] * h_3(x, y) - [\phi_2 \{f(x, y) * h_2(x, y)\} + \psi_2] \tag{B12}$$

and using the distributive law of convolution gives

$$d(x, y) = [\phi_1 \{f(x, y) * h_1(x, y)\} * h_3(x, y) + \psi_1 * h_3(x, y)] - [\phi_2 \{f(x, y) * h_2(x, y)\} + \psi_2]. \quad \text{(B13)}$$

Separating out the terms with additive constants ψ_1 and ψ_2 and re-arranging the terms with multiplicative constants ϕ_1 and ϕ_2 yields

$$d(x, y) = [f(x, y) * \{\phi_1 h_1(x, y) * h_3(x, y)\} - f(x, y) * \phi_2 h_2(x, y)] + [\psi_1 * h_3(x, y) - \psi_2]. \quad \text{(B14)}$$

and using the distributive law of convolution again results in

$$d(x, y) = f(x, y) * [\{\phi_1 h_1(x, y) * h_3(x, y)\} - \phi_2 h_2(x, y)] + [\psi_1 * h_3(x, y) - \psi_2]. \quad \text{(B15)}$$

The term in the second set of square brackets $[\psi_1 * h_3(x, y) - \psi_2]$ can be written using the two-dimensional convolution integral to give

$$\psi_1 * h_3(x, y) - \psi_2 = \left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi_1(\xi, \eta)\, h_3(x - \xi, y - \eta)\, d\xi\, d\eta\right) - \psi_2 \quad \text{(B16)}$$

and since ψ_1 does not depend on spatial position (x, y) then

$$\psi_1 * h_3(x, y) - \psi_2 = \psi_1\left(\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_3(x - \xi, y - \eta)\, d\xi\, d\eta\right) - \psi_2 \quad \text{(B17)}$$

and the integral is the volume of the PSF. It is usual to set

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h_3(\xi, \eta)\, d\xi\, d\eta = 1 \quad \text{(B18)}$$

and thus (B17) and (B18) gives

$$\psi_1 * h_3(x, y) - \psi_2 = \psi_1 - \psi_2 \quad \text{(B19)}$$

and so a constant is produced that is independent of depth. This shows the interesting result that adding constants to the images does not affect the best selected convolution ratio $h_3(x, y)$.

Now consider the first square-bracketed term in (B15), which will be denoted $\lambda(x, y)$, i.e.

$$\lambda(x, y) = \{\phi_1 h_1(x, y) * h_3(x, y)\} - \phi_2 h_2(x, y) \quad \text{(B20)}$$

and note that when $\phi_1 = \phi_2 = 1$ the term reduces to that of the original case before the linear transformation as shown in (B6). It is instructive at this point to transform the problem to the Fourier domain so that the useful property that spatial domain convolution becomes Fourier domain multiplication can be employed. Consider the Fourier transform of $\lambda(x, y)$ to give

$$\Lambda(\omega, \nu) = \phi_1 H_1(\omega, \nu) H_3(\omega, \nu) - \phi_2 H_2(\omega, \nu) \quad \text{(B21)}$$

where $\lambda(x, y) \overset{\text{FT}}{\longleftrightarrow} \Lambda(\omega, \nu)$ and $h_i(x, y) \overset{\text{FT}}{\longleftrightarrow} H_i(\omega, \nu)$ for $i = 1, 2, 3$ and assuming no noise and the correct convolution ratio was chosen the term $\Lambda(\omega, \nu)$ will reduce to zero, i.e.

$$\phi_1 H_1(\omega, \nu) H_3(\omega, \nu) - \phi_2 H_2(\omega, \nu) = 0. \quad \text{(B22)}$$

Rearranging to find the Fourier transform of the convolution ratio gives

$$H_3(\omega, \nu) = \frac{\phi_2 H_2(\omega, \nu)}{\phi_1 H_1(\omega, \nu)}. \tag{B23}$$

It is assumed that the PSF is a Gaussian for simplicity, but similar analyses could be performed for other PSF shapes. If the i^{th} 2D Gaussian PSF in the spatial domain is given by

$$h_i(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left(\frac{x^2}{\sigma_{x_i}^2} + \frac{y^2}{\sigma_{y_i}^2}\right)\right\} \tag{B24}$$

where $i = 1, 2, 3$ then its Fourier transform is given by

$$H_i(\omega, \nu) = \exp\left\{-\frac{1}{2}\left(\omega^2\sigma_{x_i}^2 + \nu^2\sigma_{y_i}^2\right)\right\} \tag{B25}$$

and thus the Fourier transform of the convolution ratio (B23) is given by

$$H_3(\omega, \nu) = \frac{\phi_2 \exp\left\{-\frac{1}{2}\left(\omega^2\sigma_{x_2}^2 + \nu^2\sigma_{y_2}^2\right)\right\}}{\phi_1 \exp\left\{-\frac{1}{2}\left(\omega^2\sigma_{x_1}^2 + \nu^2\sigma_{y_1}^2\right)\right\}}. \tag{B26}$$

which can be simplified to give

$$H_3(\omega, \nu) = \frac{\phi_2}{\phi_1} \exp\left\{-\frac{1}{2}\left(\omega^2\left(\sigma_{x_2}^2 - \sigma_{x_1}^2\right) + \nu^2\left(\sigma_{y_2}^2 - \sigma_{y_1}^2\right)\right)\right\}. \tag{B27}$$

In the implementation the convolution ratios have to be pre-computed and it is usual to make the assumption that $H_3(\omega, \nu)$ is a unit volume Gaussian PSF and so then it will be of the form

$$H_3(\omega, \nu) = \exp\left\{-\frac{1}{2}\left(\omega^2\sigma_{x_3}^2 + \nu^2\sigma_{y_3}^2\right)\right\} \tag{B28}$$

and so equating (B27) and (B28) gives

$$\exp\left\{-\frac{1}{2}\left(\omega^2\sigma_{x_3}^2 + \nu^2\sigma_{y_3}^2\right)\right\} = \frac{\phi_2}{\phi_1} \exp\left\{-\frac{1}{2}\left(\omega^2\left(\sigma_{x_2}^2 - \sigma_{x_1}^2\right) + \nu^2\left(\sigma_{y_2}^2 - \sigma_{y_1}^2\right)\right)\right\}. \tag{B29}$$

Taking natural logarithms gives

$$-\frac{1}{2}\left(\omega^2\sigma_{x_3}^2 + \nu^2\sigma_{y_3}^2\right) = \ln\left(\frac{\phi_2}{\phi_1}\right) - \frac{1}{2}\left(\omega^2\left(\sigma_{x_2}^2 - \sigma_{x_1}^2\right) + \nu^2\left(\sigma_{y_2}^2 - \sigma_{y_1}^2\right)\right). \tag{B30}$$

and separating out the terms for the orthogonal spatial frequency components ω and ν gives equations for the variances of the Gaussians of the convolution ratio as

$$\sigma_{x_3}^2 = -\frac{2}{\omega^2}\ln\left(\frac{\phi_2}{\phi_1}\right) + \left(\sigma_{x_2}^2 - \sigma_{x_1}^2\right) \tag{B31}$$

and

$$\sigma_{y_3}^2 = -\frac{2}{\nu^2}\ln\left(\frac{\phi_2}{\phi_1}\right) + \left(\sigma_{y_2}^2 - \sigma_{y_1}^2\right). \tag{B32}$$

Having completed a general monochrome analysis assuming a linear transformation of the images the next section considers the specific problem discussed in the introduction.

B.3 Colour Mixing for Depth-from-Defocus

Consider the model of the imaging and colour mixing system diagrammatically in Figure 9.2. It is assumed that the Point Spread Functions (PSFs) are identical for all three colour channels for a given camera setting. A linear transformation is applied to each colour channel following the capture and then the channels are summed to give the monochrome images that are subsequently presented to the DFD algorithm.

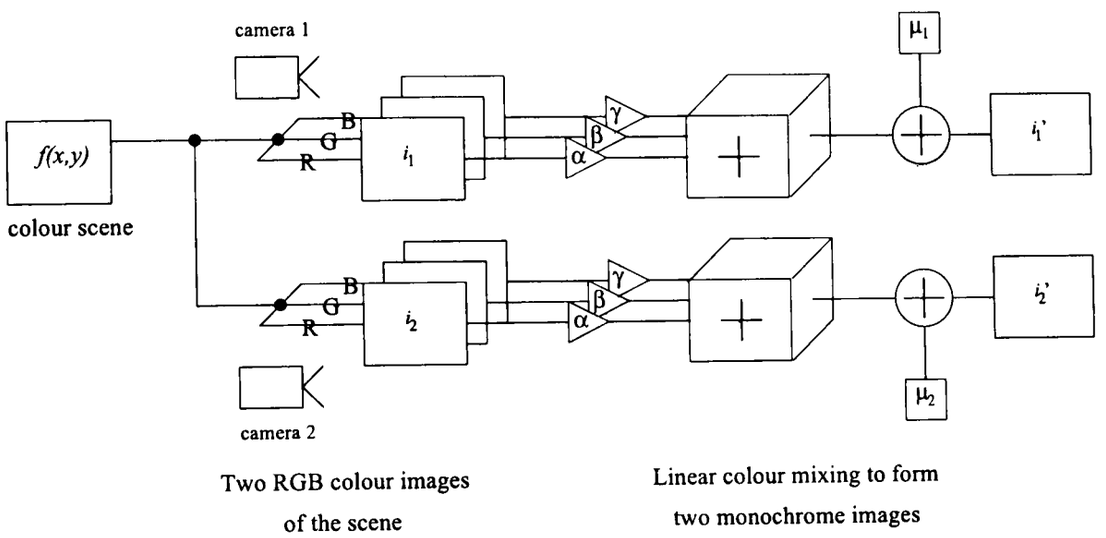


Figure B.1: Linear colour mixing model

Monochrome image i is given by

$$M_i(x, y) = \alpha_i R_i(x, y) + \beta_i G_i(x, y) + \gamma_i B_i(x, y) + \mu_i \tag{B33}$$

where $(\alpha_i, \beta_i, \gamma_i)$ are the real multiplicative constants, μ_i is an additive constant and R_i, G_i and B_i are the defocus blurred red, green and blue colour channels respectively. Two images are employed and so $i = 1, 2$. The defocused colour channels are given by

$$R_i(x, y) = f_R(x, y) * h_i(x, y) \tag{B34}$$

$$G_i(x, y) = f_G(x, y) * h_i(x, y) \tag{B35}$$

$$B_i(x, y) = f_B(x, y) * h_i(x, y) \tag{B36}$$

where f_R, f_G and f_B are the RGB colour channels of the pinhole image and h_i is the i^{th} PSFs. Expanding (B33) using (B34), (B35) and (B36) gives

$$M_i(x, y) = \alpha_i[f_R(x, y) * h_i(x, y)] + \beta_i[f_G(x, y) * h_i(x, y)] + \gamma_i[f_B(x, y) * h_i(x, y)] + \mu_i \tag{B37}$$

Two monochrome images $M_1(x, y)$ and $M_2(x, y)$ are used in Ens and Lawrence's [58] [59] DFD algorithm so that the best convolution ratio $h_3(x, y)$ is sought such that

$$\min \sum_{x,y} (M_1(x, y) * h_3(x, y) - M_2(x, y))^2 \tag{B38}$$

and in the noise-free case

$$M_1(x, y) * h_3(x, y) - M_2(x, y) = 0 \tag{B39}$$

and substituting in (B37) for $i = 1, 2$ gives

$$\begin{aligned}&(\alpha_1[f_R(x, y) * h_1(x, y)] + \beta_1[f_G(x, y) * h_1(x, y)] + \gamma_1[f_B(x, y) * h_1(x, y)] + \mu_1) * h_3 \\ &-(\alpha_2[f_R(x, y) * h_2(x, y)] + \beta_2[f_G(x, y) * h_2(x, y)] + \gamma_2[f_B(x, y) * h_2(x, y)] + \mu_2) = 0.\end{aligned} \tag{B40}$$

Separating out the RGB components gives

$$\begin{aligned}&f_R(x, y) * [\{\alpha_1 h_1(x, y) * h_3(x, y)\} - \alpha_2 h_2(x, y)] \\ &+f_G(x, y) * [\{\beta_1 h_1(x, y) * h_3(x, y)\} - \beta_2 h_2(x, y)] \\ &+f_B(x, y) * [\{\gamma_1 h_1(x, y) * h_3(x, y)\} - \gamma_2 h_2(x, y)] \\ &+[\mu_1 * h_3(x, y) - \mu_2] = 0\end{aligned} \tag{B41}$$

and from (B19) it can be seen that $\mu_1 * h_3(x, y) - \mu_2 = \mu_1 - \mu_2$ and this does not affect the optimum convolution ratio $h_3(x, y)$. If the constant is ignored for the moment and the Fourier transform of (B41) is taken then

$$F_R(\alpha_1 H_1 H_3 - \alpha_2 H_2) + F_G(\beta_1 H_1 H_3 - \beta_2 H_2) + F_B(\gamma_1 H_1 H_3 - \gamma_2 H_2) = 0 \tag{B42}$$

where $f_i(x, y) \overset{\text{FT}}{\longleftrightarrow} F_i(\omega, \nu)$ for $i = [R, G, B]$ and $h_j(x, y) \overset{\text{FT}}{\longleftrightarrow} H_j(\omega, \nu)$ for $j = 1, 2, 3$ and the spatial frequency components have been dropped for clarity. If only one colour channel existed (e.g. red) then the problem would reduce to

$$F_R(\alpha_1 H_1 H_3 - \alpha_2 H_2) = 0 \tag{B43}$$

and it was shown in (B31) and (B32) that if $\alpha_1 \neq \alpha_2$ then an offset is produced. With three colour planes the problem becomes more complicated to analyse mathematically because the contribution due to the scene does not cancel. If $\alpha_1 = \alpha_2$, $\beta_1 = \beta_2$ and $\gamma_1 = \gamma_2$ then (B42) becomes

$$\alpha F_R(H_1 H_3 - H_2) + \beta F_G(H_1 H_3 - H_2) + \gamma F_B(H_1 H_3 - H_2) = 0 \tag{B44}$$

from which it can be seen that the correct convolution ratio in the noise-free case sets $H_1 H_3 - H_2 = 0$. Thus the corresponding colour planes of both images must be scaled identically to give accurate depth estimates.

B.4 Conclusion

In the case where a monochrome image is formed from a linear combination of the colour planes, it is important that the corresponding colour planes of both images are scaled identically to give accurate depth estimates, i.e. $\alpha_1 = \alpha_2$, $\beta_1 = \beta_2$ and $\gamma_1 = \gamma_2$. It was found that the addition of the constants to each colour plane does not affect the depth returned using Ens and Lawrence's DFD algorithm.

Appendix C

HSI Analysis of Colour Mixing

C.1 Introduction

When humans discuss colour they are unlikely to specify the proportions of red, green and blue, instead they use the Hue-Saturation-Intensity (HSI) colour space without necessarily knowing it. The intensity is a measure of the brightness of the pixel and the hue gives its colour, e.g. red, yellow, green, cyan, blue, magenta etc. The saturation specifies how far the colour is from grey. This Appendix examines linear colour mixing from an HSI view-point, instead of in terms of the RGB colour space, to find out what variation an image must be possess so that colour mixing using

$$M(x, y) = \alpha R(x, y) + \beta G(x, y) + \gamma B(x, y) \tag{C1}$$

gives a different response to simply using $(\alpha, \beta, \gamma) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$.

C.2 RGB to HSI Transformation

The transformation from RGB to HSI begins by normalising the pixels to lie in the closed interval [0, 1]. The value or intensity of the red, green and blue pixels is denoted R, G and B respectively. The hue H is given by

$$H = \begin{cases} \theta & \text{if } B \le G \\ 2\pi - \theta & \text{if } B > G \end{cases} \tag{C2}$$

where the angle θ (in radians) is given by

$$\theta = \cos^{-1}\left(\frac{\frac{1}{2}[(R-G)+(R-B)]}{\left[(R-G)^2 + (R-B)(G-B)\right]^{\frac{1}{2}}} \right). \tag{C3}$$

The hue is often normalised to lie in the range [0, 1] by dividing by 2π. The saturation S is given by

$$S = 1 - \frac{3}{R+G+B}\min(R, G, B) \tag{C4}$$

where $\min(R, G, B)$ is a non-linear function that returns the lowest pixel value of the red, green and blue pixels. The intensity I is the only linear function in the transformation and it is given by

$$I = \frac{R+G+B}{3}. \tag{C5}$$

The inverse transformation of the HSI coordinates to RGB is less straightforward and it depends on the value of the hue, which is assumed to be in the range $[0, 2\pi]$. If $0 \le H < \frac{2\pi}{3}$ (the RG sector) then the RGB components are given by

$$R = I\left(1 + \frac{S\cos H}{\cos\left(\frac{\pi}{3} - H\right)}\right) \tag{C6}$$

$$G = 1 - (R + B) \tag{C7}$$

$$B = I(1 - S) \tag{C8}$$

If $\frac{2\pi}{3} \le H < \frac{4\pi}{3}$ (the GB sector) then the hue must be modified to $H \to H - \frac{2\pi}{3}$ and then

$$R = I(1 - S) \tag{C9}$$

$$G = I\left(1 + \frac{S\cos H}{\cos\left(\frac{\pi}{3} - H\right)}\right) \tag{C10}$$

$$B = 1 - (R + G) \tag{C11}$$

and if $\frac{4\pi}{3} \le H < 2\pi$ (the BR sector) then the hue must be modified with $H \to H - \frac{4\pi}{3}$ and then

$$R = 1 - (G + B) \tag{C12}$$

$$G = I(1 - S) \tag{C13}$$

$$B = I\left(1 + \frac{S\cos H}{\cos\left(\frac{\pi}{3} - H\right)}\right). \tag{C14}$$

Colour mixing was formulated in the RGB space, but it is useful to consider whether it could be used on images that only vary in hue, saturation or intensity, where the remaining two quantities are a constant. More generally the equations can be written as

$$X_1 = I(I - S) \tag{C15}$$

$$X_2 = I\left(1 + \frac{S\cos H}{\cos\left(\frac{\pi}{3} - H\right)}\right) \tag{C16}$$

$$X_3 = 1 - (X_1 + X_2) \tag{C17}$$

and then the following table could be used to determine which equation applies for RGB depending on the sector.

Table C.1: Variable for a given sector

Variable	R-G Sector	G-B Sector	B-R Sector
X_1	B	R	G
X_2	R	G	B
X_3	G	B	R

C.3 HSI Colour Mixing Analysis

The HSI transformation equations are useful to analyse colour mixing and then find the effects of allowing only one of the HSI components to vary. The monochrome image $M(x, y)$ used for DFD is given by (C1). It can be shown that the particular sector of hue is irrelevant in the conclusions formed and for the derivations the R-G sector will be used. The general colour mixing equations are derived below and then for each specific case the general equations are altered. The spatial location of the pixel (x, y) must be included to give

$$R(x, y) = I(x, y)\left(1 + \frac{S(x, y)\cos H(x, y)}{\cos\left(\frac{\pi}{3} - H(x, y)\right)}\right) \tag{C18}$$

$$G(x, y) = 1 - (R(x, y) + B(x, y)) \tag{C19}$$

$$B(x, y) = I(x, y)(1 - S(x, y)) \tag{C20}$$

The green colour plane $G(x, y)$ needs to be written in terms of HSI so the red and blue plane equations are substituted in to give

$$G(x, y) = 1 - \left(I(x, y)\left(1 + \frac{S(x, y)\cos H(x, y)}{\cos\left(\frac{\pi}{3} - H(x, y)\right)}\right) + I(x, y)(1 - S(x, y))\right) \tag{C21}$$

and re-arranging gives

$$G(x, y) = 1 - I(x, y)\left(2 + \frac{S(x, y)\cos H(x, y)}{\cos\left(\frac{\pi}{3} - H(x, y)\right)} - S(x, y)\right). \tag{C22}$$

Using the HSI-based equations gives the colour mixed monochrome image as

$$M(x,y) = \alpha\left(I(x,y)\left(1 + \frac{S(x,y)\cos H(x,y)}{\cos\left(\frac{\pi}{3} - H(x,y)\right)}\right)\right) +$$
$$\beta\left(1 - \left(I(x,y)\left(1 + \frac{S(x,y)\cos H(x,y)}{\cos\left(\frac{\pi}{3} - H(x,y)\right)}\right) + I(x,y)\,(1 - S(x,y))\right)\right) + \tag{C23}$$
$$\gamma\,(I(x,y)\,(1 - S(x,y))).$$

Now consider the case where two of the components of the HSI are held constant and the remaining component is allowed to change spatially.

C.3.1 Hue Variation and Colour Mixing

Consider a surface with a varying hue $H(x, y)$ and a constant saturation S and intensity I. The resulting monochrome image can be found from (C23) and is given by

$$M(x,y) =$$
$$\alpha\, I\left(1 + \frac{S\cos H(x,y)}{\cos\left(\frac{\pi}{3} - H(x,y)\right)}\right) + \beta\left(1 - I\left(2 + \frac{S(x,y)\cos H(x,y)}{\cos\left(\frac{\pi}{3} - H(x,y)\right)} - S\right)\right) + \gamma\, I\,(1 - S). \tag{C24}$$

Rearranging the terms together gives

$$M(x,y) = [\alpha\, I + \beta - 2\, I\,\beta + \beta\, I\, S + \gamma\, I(1 - S)] + \frac{I\, S\cos H(x,y)}{\cos\left(\frac{\pi}{3} - H(x,y)\right)}\,(\alpha - \beta) \tag{C25}$$

and denoting the constant term in square brackets as C gives

$$M(x,y) = C + \frac{I\, S\cos H(x,y)}{\cos\left(\frac{\pi}{3} - H(x,y)\right)}\,(\alpha - \beta). \tag{C26}$$

The resulting monochrome image $M(x, y)$ is not proportional to the intensity I and hence the image has been changed through mixing using (α, β, γ).

C.3.2 Saturation Variation and Colour Mixing

Consider a image that has a varying saturation $S(x, y)$ and a constant hue H and intensity I. The colour mixed image $M(x, y)$ is thus given by modifying (C23) to give

$$M(x,y) = \alpha\left(I + \frac{I\, S(x,y)\cos H}{\cos\left(\frac{\pi}{3} - H\right)}\right) + \beta\left(1 - 2\, I - \frac{I\, S(x,y)\cos H}{\cos\left(\frac{\pi}{3} - H\right)} + I\, S(x,y)\right) +$$
$$\gamma\; I\,(1 - S(x,y)). \tag{C27}$$

and collecting the terms gives

$$M(x,y) = I(\alpha - 2\,\beta + \gamma) + \frac{I\, S(x,y)\cos H}{\cos\left(\frac{\pi}{3} - H\right)}\,(\alpha - \beta) + I\, S(x,y)\,(\beta - \gamma) + \beta. \tag{C28}$$

If $\alpha - \beta = 0$ and $\beta - \gamma = 0$, which implies $\alpha = \beta = \gamma$, then the image reduces to a constant intensity, given by

$$M(x, y) = I(\alpha - 2\beta + \gamma) + \beta \tag{C29}$$

and otherwise the resulting monochrome image $M(x, y)$ does not have a constant intensity and is dependent on the varying saturation $S(x, y)$ as can be seen by rearranging (C28) to give

$$M(x, y) = [I(\alpha - 2\beta + \gamma) + \beta] + S(x, y)\left[\frac{I\cos H}{\cos\left(\frac{\pi}{3} - H\right)}(\alpha - \beta) + I(\beta - \gamma)\right]. \tag{C30}$$

The terms in square brackets are constants and will be denoted C_1 and C_2 so that

$$M(x, y) = C_1 + C_2 S(x, y) \tag{C31}$$

Depending on the sign of C_2 the saturation term $S(x, y)$ can either increase or decrease the brightness of the colour mixed monochrome image.

C.3.3 Intensity Variation and Colour Mixing

Now consider a surface that has a changing intensity $I(x, y)$ but a constant hue H and saturation S. The general equation (C23) becomes

$$\begin{aligned} M(x, y) = \alpha\left(I(x, y)\left(1 + \frac{S\cos H}{\cos\left(\frac{\pi}{3} - H(x, y)\right)}\right)\right) + \\ \beta\left(1 - I(x, y)\left(2 + \frac{S\cos H}{\cos\left(\frac{\pi}{3} - H\right)} - S(x, y)\right)\right) + \gamma I(x, y)(1 - S) \end{aligned} \tag{C32}$$

and rearranging gives

$$M(x, y) = I(x, y)\left[(\alpha - 2\beta + \gamma) + \frac{S\cos H}{\cos\left(\frac{\pi}{3} - H\right)}(\alpha - \beta) + S(\beta - \gamma)\right] \tag{C33}$$

and since the term in square brackets is a constant then $M(x, y) \propto I(x, y)$, thus showing that colour mixing has not been performed. The consequence of this derivation is that an image with a constant hue and saturation but a varying intensity cannot be colour mixed to yield a different intensity image from using $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$.

C.4 Conclusion

The analysis has shown that if the hue or the saturation vary with spatial position then colour mixing can be applied, but if only the intensity changes and the hue and saturation remain constant then colour mixing is no different from using the monochrome case of $\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$ except for a scaling factor.

Appendix D

Gaussian Convolution Ratios

D.1 Introduction

Ens and Lawrence's [58] [59] DFD algorithm relates the point spread functions (PSFs) of cameras 1 and 2, denoted $h_1(x, y)$ and $h_2(x, y)$, through a function known as the *convolution ratio* $h_3(x, y)$, given by

$$h_1(x, y) * h_3(x, y) = h_2(x, y). \tag{D1}$$

If the PSFs are Gaussian functions then it is known that the convolution ratio is also a Gaussian. This Appendix derives the relationship between the spreads of the PSFs and the convolution ratio.

D.2 Derivation of the Convolution Ratio

A 1-D Gaussian centred on $x = 0$ is given by

$$h_{i_x}(x) = \frac{1}{\sqrt{2\pi}\,\sigma_{i_x}} \exp\left\{-\frac{1}{2}\,\frac{x^2}{\sigma_{i_x}^2}\right\} \tag{D2}$$

where σ_{i_x} is the standard deviation of the Gaussian. The 2-D Gaussian is a separable function and it is given by

$$h_i(x, y) = h_{i_x}(x)\, h_{i_y}(y) \tag{D3}$$

$$h_i(x, y) = \frac{1}{\sqrt{2\pi}\,\sigma_{i_x}} \exp\left\{-\frac{1}{2}\,\frac{x^2}{\sigma_{i_x}^2}\right\} \frac{1}{\sqrt{2\pi}\,\sigma_{i_y}} \exp\left\{-\frac{1}{2}\,\frac{y^2}{\sigma_{i_y}^2}\right\} \tag{D4}$$

$$\Rightarrow h_i(x, y) = \frac{1}{2\pi\sigma_{i_x}\sigma_{i_y}} \exp\left\{-\frac{1}{2}\left(\frac{x^2}{\sigma_{i_x}^2} + \frac{y^2}{\sigma_{i_y}^2}\right)\right\}. \tag{D5}$$

The 2-D Continuous-Space Fourier Transform (CSFT) of a function $f(x, y)$ is given by

$$F(\Omega_x, \Omega_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)\, e^{-j(\Omega_x x + \Omega_y y)}\, dx\, dy \tag{D6}$$

where Ω_x and Ω_y are the frequency components and $j = \sqrt{-1}$. If $f(x, y)$ is separable then it can be rewritten as $f(x, y) = f_x(x)\, f_y(y)$ and the 2-D CSFT becomes

$$F(\Omega_x, \Omega_y) = \left(\int_{-\infty}^{\infty} f_x(x)\, e^{-j\Omega_x x}\, dx\right)\left(\int_{-\infty}^{\infty} f_y(y)\, e^{-j\Omega_y y}\, dy\right) = F(\Omega_x)\, F(\Omega_y). \tag{D7}$$

Now consider the 1-D CSFT of a Gaussian function $h_{i_x}(x)$, given by

$$H_{i_x}(\Omega_x) = \int_{-\infty}^{\infty} h_{i_x}(x)\, e^{-j\Omega_x x}\, dx \tag{D8}$$

and substituting in (D2) gives

$$H_{i_x}(\Omega_x) = \frac{1}{\sqrt{2\pi}\,\sigma_{i_x}} \int_{-\infty}^{\infty} \exp\left\{-\frac{1}{2}\,\frac{x^2}{\sigma_{i_x}^2}\right\} \exp\{-j\,\Omega_x\, x\}\, dx \tag{D9}$$

and this can be simplified to

$$H_{i_x}(\Omega_x) = \exp\left\{-\frac{1}{2}\,\Omega_x^2\,\sigma_{i_x}^2\right\}. \tag{D10}$$

Thus, the CSFT of the separable 2-D Gaussian is given by

$$H_i(\Omega_x, \Omega_y) = \exp\left\{-\frac{1}{2}\,\Omega_x^2\,\sigma_{i_x}^2\right\} \exp\left\{-\frac{1}{2}\,\Omega_y^2\,\sigma_{i_y}^2\right\} \tag{D11}$$

$$\Rightarrow H_i(\Omega_x, \Omega_y) = \exp\left\{-\frac{1}{2}\left(\Omega_x^2\,\sigma_{i_x}^2 + \Omega_y^2\,\sigma_{i_y}^2\right)\right\}. \tag{D12}$$

The CSFT of the convolution equation (D1) is given by

$$h_1(x, y) * h_3(x, y) = h_2(x, y) \overset{\text{CSFT}}{\longleftrightarrow} H_1(\Omega_x, \Omega_y)\, H_3(\Omega_x, \Omega_y) = H_2(\Omega_x, \Omega_y) \tag{D13}$$

as convolution in the spatial domain becomes multiplication in the Fourier frequency domain. Thus,

$$\exp\left\{-\frac{1}{2}\left(\Omega_x^2\,\sigma_{1_x}^2 + \Omega_y^2\,\sigma_{1_y}^2\right)\right\} \exp\left\{-\frac{1}{2}\left(\Omega_x^2\,\sigma_{3_x}^2 + \Omega_y^2\,\sigma_{3_y}^2\right)\right\} = \exp\left\{-\frac{1}{2}\left(\Omega_x^2\,\sigma_{2_x}^2 + \Omega_y^2\,\sigma_{2_y}^2\right)\right\} \tag{D14}$$

$$\Rightarrow \exp\left\{-\frac{1}{2}\left(\Omega_x^2\left[\sigma_{1_x}^2 + \sigma_{3_x}^2\right] + \Omega_y^2\left[\sigma_{1_y}^2 + \sigma_{3_y}^2\right]\right)\right\} = \exp\left\{-\frac{1}{2}\left(\Omega_x^2\,\sigma_{2_x}^2 + \Omega_y^2\,\sigma_{2_y}^2\right)\right\} \tag{D15}$$

and thus for equality to hold

$$\sigma_{1_x}^2 + \sigma_{3_x}^2 = \sigma_{2_x}^2 \tag{D16}$$

$$\sigma_{1_y}^2 + \sigma_{3_y}^2 = \sigma_{2_y}^2. \tag{D17}$$

A simple rearrangement shows that the required spread in the x and y-directions for the convolution ratio is given by

$$\sigma_{3_x}^2 = \sigma_{2_x}^2 - \sigma_{1_x}^2 \tag{D18}$$

$$\sigma_{3_y}^2 = \sigma_{2_y}^2 - \sigma_{1_y}^2. \tag{D19}$$

Appendix E

Checkerboard Experiments

E.1 Introduction

Images of a checkerboard pattern perpendicular to the camera's optical axis were obtained using the Basler A631fc colour camera and the 24mm Sigma photographic lens for six equally spaced depths between 0.423m and 0.673m with three different apertures: f/2.8; f/4; and f/5.6. The images were processed using the MATLAB script developed to implement DFD. Experiments were performed using image window sizes (W_I) of 32×32 and 64×64 pixels. The convolution ratio window (W_{CR}) was allowed to change with depth or it was fixed at 21×21, 31×31, 41×41 or 51×51 pixels for a 64×64 image window. For a 32×32 image window only the fixed convolution ratio window of 21×21 could be used.

The dimensions of the squares of the checkerboard were measured for each distance and then an ideal focused checkerboard image was created that was subsequently defocus blurred using the PSF data for the camera. This was used as a visual check and as a method of checking the parameters in a noise-free environment, except for the ubiquitous quantisation noise.

E.2 Results and Analysis

The first noticeable feature of the results from the experiment was that the usable depth range was about 0.414m to 0.523m. Although PSFs were calculated up to 0.725m, the significant defocus blurring reduces the variance of the imaged texture and consequently the signal-to-noise ratio decreases. A further effect of a large depth range is that the image overlap problem gets progressively worse. Although the range is small at 0.109m, it is larger than that used in 2 of the 18 compared in Section 2.6. For the purposes of testing the colour mixing algorithms it is sufficient. Different camera parameters or changing the focus position instead of the f-number (thus increasing the depth sensitivity [74]) could be used to improve the range for a specific application.

The results in Table E.1 show the mean depth error and the variance of the depth error for checkerboards at 0.423m, 0.473m and 0.523m in simulation (S) and practice (P) for the three different error measures.

For a fixed convolution ratio window of 21×21 an image window of 32×32 had a lower mean error than using 64×64, but the variance of the error was four times greater.

In practice there was very little difference in the mean depth error using convolution window sizes of 21×21, 31×31, 41×41 or 51×51 pixels. However, the variance of the error steadily increased with increasing window size. This effect is attributed to the fact that as the convolution ratio window increases in size the result of the restricted convolution $\hat{i}_2(x, y) = i_1(x, y) * h_3(x, y)$ gets smaller and consequently so does $i_2(x, y)$ and thus less image data is employed, which makes it more prone to errors due to noise.

The variable convolution ratio window size produced worse results than a fixed window, and so this method can be eliminated. For example, with a 32×32 image window the mean error using a variable convolution ratio window was 1.3 times larger and the variance 2.4 times greater. Of the fixed convolution ratio window sizes the L_2-norm produced better results overall than the L_1-norm and both performed much better than the Information-Divergence measure.

Table E.1: Mean error and variance (in brackets) using different error measures for f/5.6 and f/2.8

Window Size			Error Measure		
W_I	W_{CR}	S / P	L_2	L_1	I-Divergence
32×32	Variable	P	5.33 (40.5)	101 (4.66)	7.00 (46.7)
		S	1.00 (9.48)	1.67 (11.2)	14.0 (51.0)
32×32	21×21	P	-4.00 (17.2)	-3.67 (17.9)	32.3 (102)
		S	-0.333 (1.50)	-0.333 (1.57)	42.0 (106)
64×64	Variable	P	11.3 (29.5)	76.3 (52.4)	29.7 (79.7)
		S	0 (0.036)	0 (0.0363)	68.0 (87.2)
64×64	21×21	P	-6.00 (2.95)	-7.00 (4.07)	-10.0 (7.76)
		S	-0.333 (0)	0.333 (0)	9.33 (23.8)
64×64	31×31	P	-6.00 (3.67)	-7.33 (4.64)	9.67 (49.6)
		S	0 (0.0897)	0 (0.093)	4.00 (25.4)
64×64	41×41	P	-6.67 (4.08)	-7.00 (5.07)	12.7 (67.8)
		S	0 (0.143)	0 (0.143)	22.3 (69.8)
64×64	51×51	P	-5.00 (11.7)	-5.33 (12.0)	31.0 (99.5)
		S	-0.333 (1.40)	-0.333 (1.41)	42.3 (107)

The mean error in the simulation results (denoted S) with only quantisation noise present were very low using a 64×64 window and as with the practical experiments, the standard deviation of the error increased with increasing convolution ratio window size.

As the L_2-norm worked the best it was employed in the next set of tests, shown in Table E.2, where all three aperture combinations were tested.

Table E.2: Mean error and variance (in brackets) using the L_2-norm

Window Size			Aperture Combination		
W_I	W_{CR}	S / P	f/5.6, f/2.8	f/5.6, f/4	f/4, f/2.8
32×32	Variable	P	5.33 (40.5)	93.0 (85.2)	109 (92.3)
		S	1.00 (9.48)	4.33 (18.2)	0.667 (9.34)
32×32	21×21	P	-4.00 (17.2)	5.00 (16.7)	-30.7 (26.3)
		S	-0.333 (1.50)	-0.667 (1.71)	-0.333 (2.60)
64×64	Variable	P	11.3 (29.5)	1.67 (3.18)	-29.7 (4.88)
		S	0 (0.036)	0 (0.120)	-0.333 (0.363)
64×64	21×21	P	-6.00 (2.95)	4.33 (2.74)	-33.7 (3.00)
		S	-0.333 (0)	0 (0.120)	-0.667 (0.153)
64×64	31×31	P	-6.00 (3.67)	4.33 (3.23)	-32.0 (0.470)
		S	0 (0.0897)	0 (0)	-0.667 (0.275)
64×64	41×41	P	-6.67 (4.08)	4.33 (3.51)	-34.3 (5.70)
		S	0 (0.143)	0 (0.140)	-0.333 (0.440)
64×64	51×51	P	-5.00 (11.7)	4.33 (11.1)	-33.0 (15.2)
		S	-0.333 (1.40)	-0.333 (1.21)	-0.333 (2.14)

Overall it was found that the aperture combination of ($f_1 = 5.6$, $f_2 = 4$) produced the best results and those produced using ($f_1 = 4$, $f_2 = 2.8$) were the worst.

E.3 Slope Experiments

A colour checkerboard was pasted to a slope that was between 0.440m and 0.520m from the camera. The three different aperture combinations were tested along with the three different error measures. The 32×32 image window was employed and the convolution ratio window was fixed at 21×21. The depth error was measured at 2745 equally spaced points in the images and the results are presented in Tables E.3 to E.5.

In Section 6.8.2 an improved normalisation equation is shown to compensate for the exposure change. The results are shown in Tables D.3 to D.5 in brackets.

Table E.3: MSE results for checkerboard pattern

	Mean Square Error / mm^2		
Apertures	L_1-norm	L_2-norm	I-Divergence
f/5.6, f/2.8	0.425 (0.232)	0.374 (0.227)	19.2 (1.49)
f/5.6, f/4	0.373 (0.170)	0.327 (0.168)	23.3 (1.54)
f/4, f/2.8	2.27 (1.91)	2.22 (1.92)	24.2 (3.10)

Table E.4: Mean error results for checkerboard pattern

	Mean Error / mm		
Apertures	L_1-norm	L_2-norm	I-Divergence
f/5.6, f/2.8	-4.97 (-5.27)	-4.72 (-5.01)	68.5 (1.01)
f/5.6, f/4	3.94 (3.76)	4.24 (3.94)	81.8 (8.47)
f/4, f/2.8	-31.6 (-32.1)	-32.0 (-32.1)	83.6 (-11.3)

Table E.5: Variance of the error results for checkerboard pattern

	Variance of Error / mm^2		
Apertures	L_1-norm	L_2-norm	I-Divergence
f/5.6, f/2.8	0.400 (0.205)	0.352 (0.201)	14.5 (1.49)
f/5.6, f/4	0.357 (0.156)	0.310 (0.152)	16.7 (1.47)
f/4, f/2.8	1.27 (0.880)	1.19 (0.882)	17.2 (2.98)

The best error measure was found to be the L_2-norm, as used by Ens and Lawrence [58] [59], and it is noticeable that there is very little difference in the errors using the aperture combinations ($f_1 = 5.6$, $f_2 = 2.8$) and ($f_1 = 5.6$, $f_2 = 4$). The aperture combination ($f_1 = 4$, $f_2 = 2.8$) produced much worse results with all three measures. As both images were fairly defocused with this setting, it is assumed that the overall information content is less than the other two combinations.

By using the improved normalisation the MSE using the I-Divergence decreased by 11.9 times on average where as the decrease was only 1.6 times using the L_2-norm and 1.5 times using the L_1-norm. Thus, the I-Divergence measure is particularly sensitive to noise in the images and the relative scaling of $\hat{i}_2(x, y)$ and $i_2(x, y)$.

E.4 Conclusion

To ensure good localisation, a 32×32 image window is preferable to a 64×64 window. With a 32×32 window a fixed convolution ratio window size of 21×21 performed better than using a variable window size. The accuracy of the results were similar using ($f_1 = 5.6$, $f_2 = 2.8$) and ($f_1 = 5.6$, $f_2 = 4$) compared to ($f_1 = 4$, $f_2 = 2.8$).

Appendix F

Analysis of a Step in Depth

F.1 Introduction

Ens and Lawrence's algorithm is based on the equifocal assumption that the depth is constant within a window and for a real scene this assumption is generally violated. If the scene $f(x, y)$ has a varying depth then the PSF $h_k(x, y, \xi, \eta)$ is space-varying and the defocused image is given by

$$i_k(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi, \eta)\, h_k(x, y, \xi, \eta)\, d\xi\, d\eta. \tag{F1}$$

If it is assumed that the depth is constant then the integral can be reduced to the convolution integral, given by

$$i_k(x, y) = \int_{\infty}^{\infty}\int_{-\infty}^{\infty} f(\xi, \eta)\, h_k(x-\xi, y-\eta)\, d\xi\, d\eta \tag{F2}$$

Ens and Lawrence's algorithm searches for the best convolution ratio such that the less defocused image, image $i_1(x, y)$ convolved with a convolution ratio $h_3(x, y)$ from the look-up table approximates the more defocused image $i_2(x, y)$. This section considers the simple case of a step in depth and the effect that assuming space-invariance has.

F.2 Analysis

Consider a scene $f(x, y)$ composed of two regions that will be denoted A and B. The unit step is denoted $u(x)$ and is given by

$$u(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \tag{F3}$$

and so the scene can be written as

$$f(x, y) = f_A(x, y)\, u(-x) + f_B(x, y)\, u(x)\,. \tag{F4}$$

If the regions A and B are defined to have zero intensity outside their required support then (F4) can be simplified to

$$f(x, y) = f_A(x, y) + f_B(x, y) \tag{F5}$$

The image of the defocused scene $i_k(x, y)$ can be written using the sum of two convolutions because each region has a constant depth, thus

$$i_k(x, y) = f_A(x, y) * h_{k_A}(x, y) + f_B(x, y) * h_{k_B}(x, y). \tag{F6}$$

Due to the linearity property of the Fourier transform the discrete Fourier transform of (F4) is given by

$$I_k(u, v) = F_A(u, v)\, H_{k_A}(u, v) + F_B(u, v)\, H_{k_B}(u, v). \tag{F7}$$

and if $H_{k_A} \neq H_{k_B}$, i.e. the regions are at different depths, then with two defocused images there are four PSFs. If the depths of the regions are the same then () becomes

$$I_k(u, v) = [F_A(u, v) + F_B(u, v)]\, H_k(u, v) = F(u, v)\, H_k(u, v). \tag{F8}$$

In the case where the image region is at a constant depth the convolution ratio is given by

$$H_3(u, v) = \frac{F(u, v)\, H_2(u, v)}{F(u, v)\, H_1(u, v)} = \frac{H_2(u, v)}{H_1(u, v)} \tag{F9}$$

but in the case where there is a step in the depth the convolution ratio is given by

$$H_3(u, v) = \frac{F_A(u, v)\, H_{2_A}(u, v) + F_B(u, v)\, H_{2_B}(u, v)}{F_A(u, v)\, H_{1_A}(u, v) + F_B(u, v)\, H_{1_B}(u, v)} \tag{F10}$$

Suppose the region of interest is region A then the error in the convolution ratio is given by the difference between the required convolution ratio due to A and that due to the step in the depth with regions A and B, i.e.

$$H_3(u, v) = \frac{F_A(u, v)\, H_{2_A}(u, v)}{F_A(u, v)\, H_{1_A}(u, v)} - \frac{F_A(u, v)\, H_{2_A}(u, v) + F_B(u, v)\, H_{2_B}(u, v)}{F_A(u, v)\, H_{1_A}(u, v) + F_B(u, v)\, H_{1_B}(u, v)} \tag{F11}$$

If the contribution of region B to the convolution ratio can be removed using colour mixing then the depth estimate will be more accurate. In the next section a couple of simulation results are presented to show the effect of the step.

F.3 Simulations

The set-up of the simulation is shown diagrammatically in Figure F.1 where the top step was moved from 0.48m to 0.76m in 4cm steps. The experimentally-derived PSF data from the 16mm video lens was used as the results were produced during the earlier stages of the research. The colour image were converted to monochrome using an equal weighting of the colour planes.

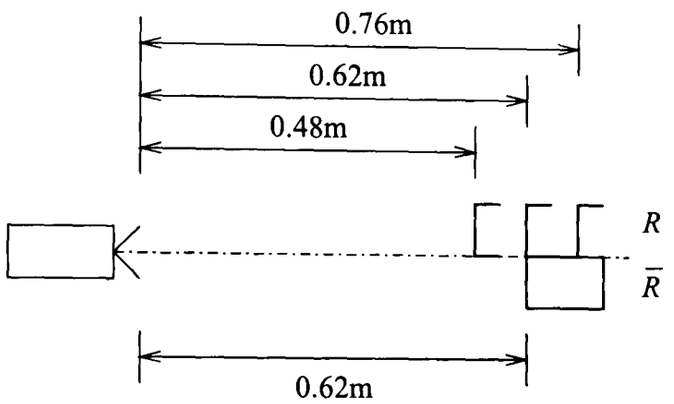


Figure F.1: The set-up for the simulation experiment where the top step is moved in small increments

In the first experiment images of grass taken with a high resolution digital stills camera were used to create a texture. Actual PSF data was used to simulate the defocus blurring of the texture on the two steps. The right hand step was held constant at a depth of 0.62m and the left hand step was varied in depth. All pixels running down the boundary edge were processed and then the mean and variance of the depths were calculated. Figure F.2 below shows the mean and variance of the step depth as a function of the actual depth of the left hand step. The dotted line shows the depth that would be obtained if the mean depth was equal to the actual depth. The right hand figure shows the depth error, which appears to show a fairly linear relationship with depth. Also plotted is the depth that was obtained if the height of the right hand step was equal to that of the left, thus providing a benchmark and the results are denoted *w/o step* in the figure legends.

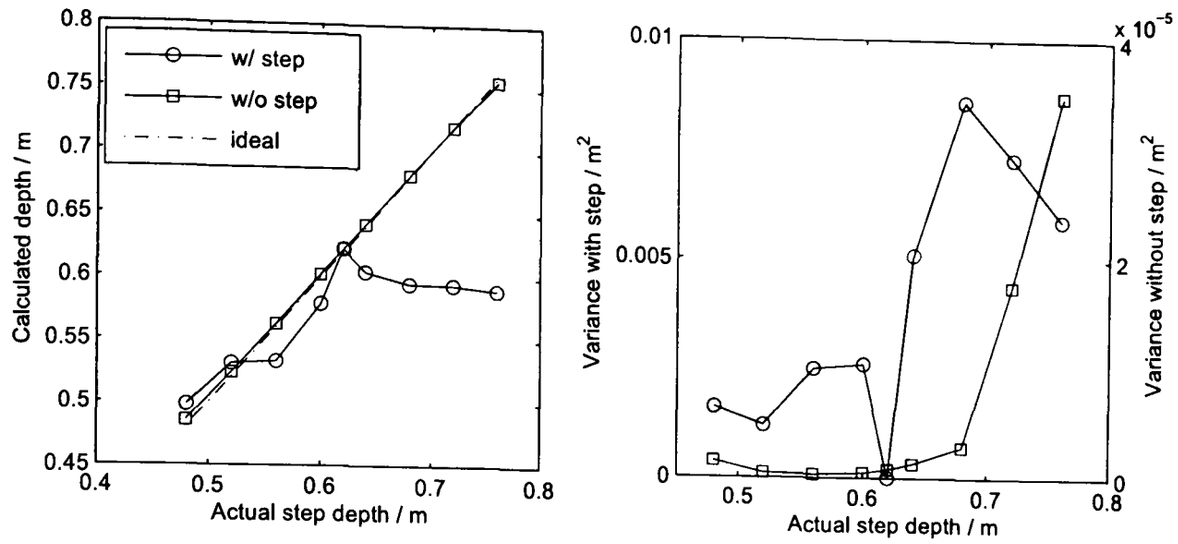


Figure F.2: The results for grass / grass combination

The second experiment was done to ensure the results were not dependent on the same texture being used on the top and bottom of the steps. Grass and carpet were used on the upper and lower steps respectively and the results are shown in Figure F.3.

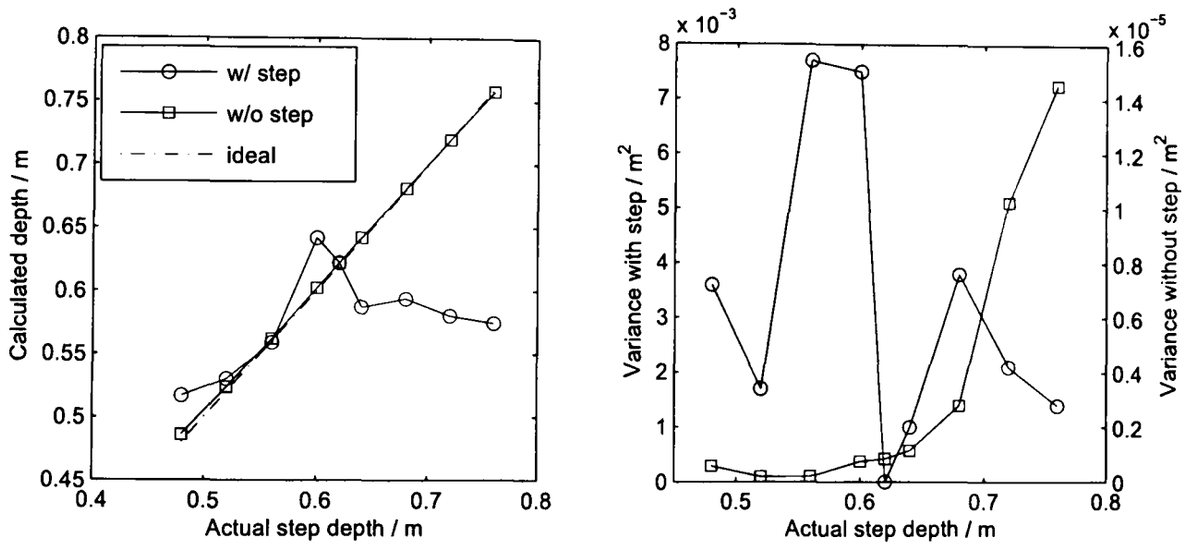


Figure F.3: The results for grass / carpet combination

When the region of interest R is closer to the camera's focus position than $\overline{R}$ it possesses more power in the high frequency components compared to those pixels in $\overline{R}$. When there is no difference in the depth of the steps the depth estimates are the same as for the control case. The worst depth errors occur when the region R is further from the camera than $\overline{R}$, which is where region R possesses less high frequency power. As the mean depth estimates are consistently poor it is likely that the high frequency components of $\overline{R}$ are adversely affecting the DFD algorithm.

Lai *et al.* [67] stated that from the Gaussian lens law and plane geometry that the depth measured at the edge of a depth discontinuity is the depth due to the nearer side.

F.4 Conclusion

This appendix has considered the effect of a step in depth in an image segment and the results have shown that the texture of the object closer to the camera dominated. If the region of interest is closer to the camera than the other region then the depth error is not significant. However, if the situation is reversed and depth error becomes significant. This work has highlighted the problem of object boundaries.

Appendix G

Colour Image Textures

G.1 Introduction

For the purposes of testing the colour mixing algorithm it was useful to have a variety of colour textures available. Images of size 2560×1920 pixels were captured and saved in uncompressed Tiff mode using a 5 megapixel Panasonic DMC-FZ20 digital stills camera. An uncompressed file format was used to ensure that the image quality was not degraded. The next section shows the 27 textures that were captured and used during the research.

G.2 Images Captured

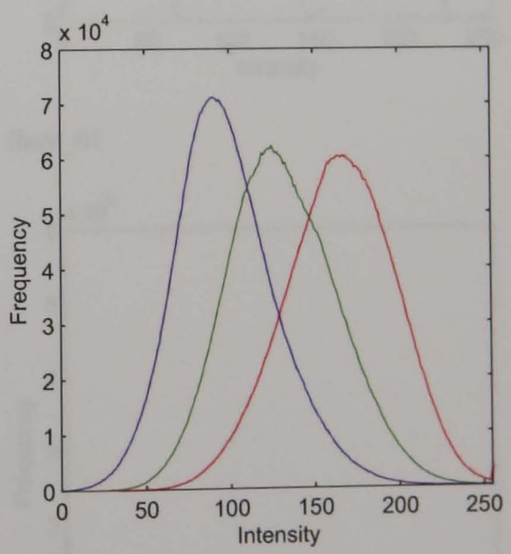


Figure G.1: carpet_01

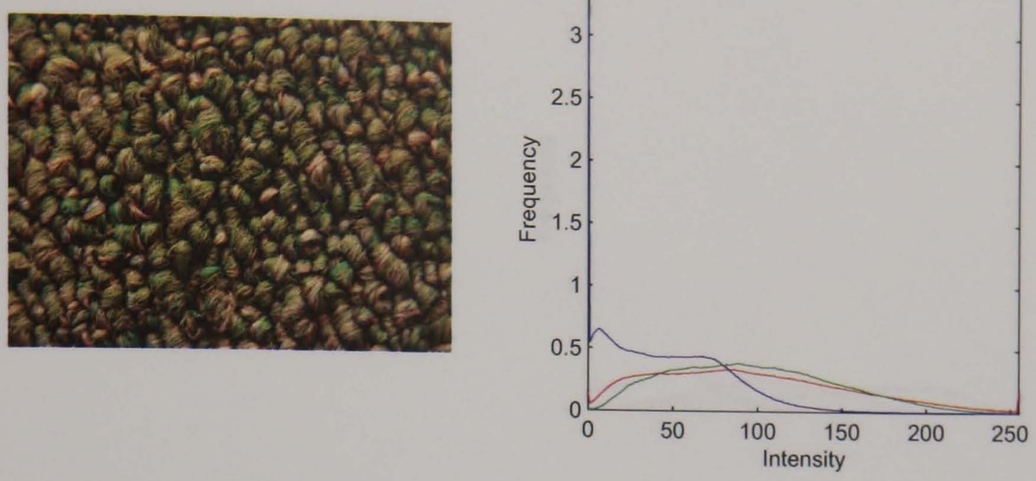


Figure G.2: carpet_02

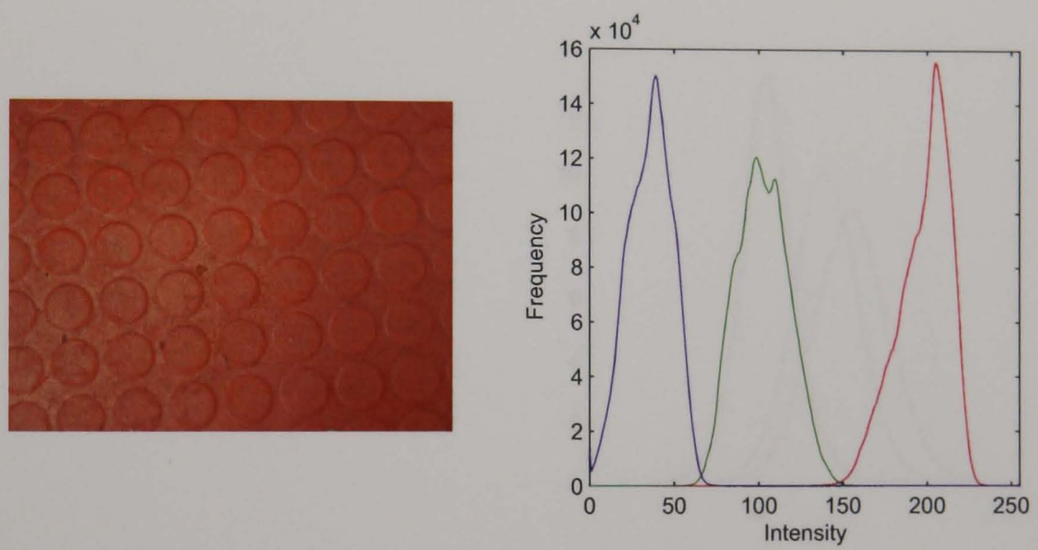


Figure G.3: floor_01

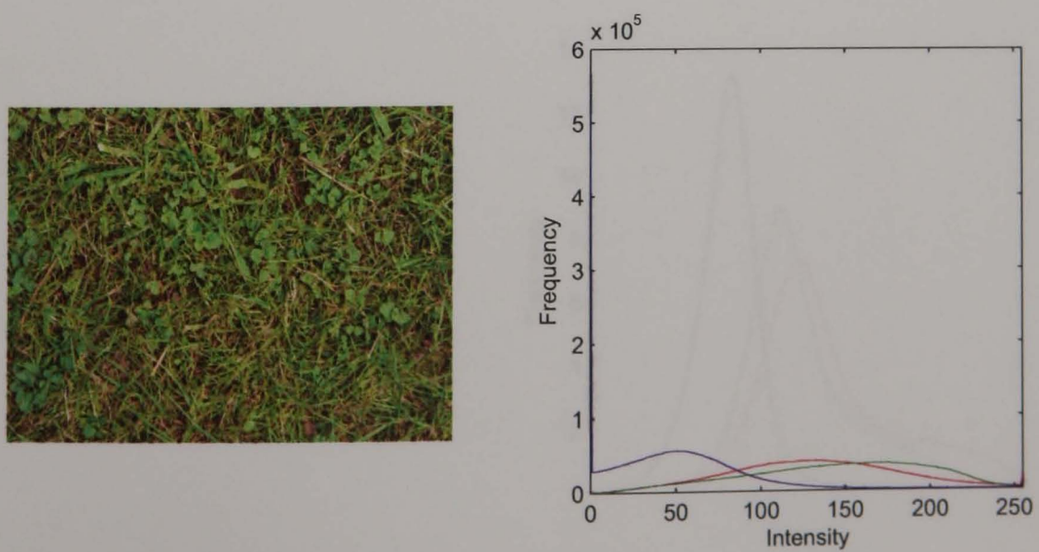


Figure G.4: grass_01

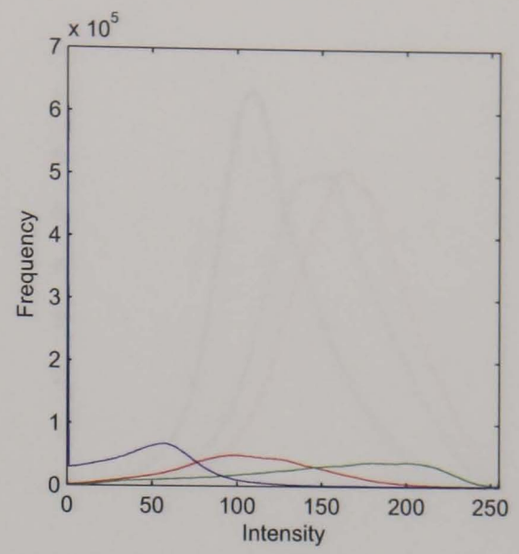


Figure G.5: grass_02

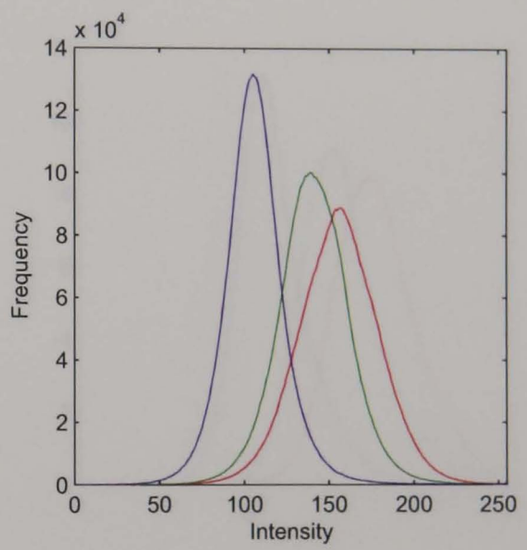


Figure G.6: road_01

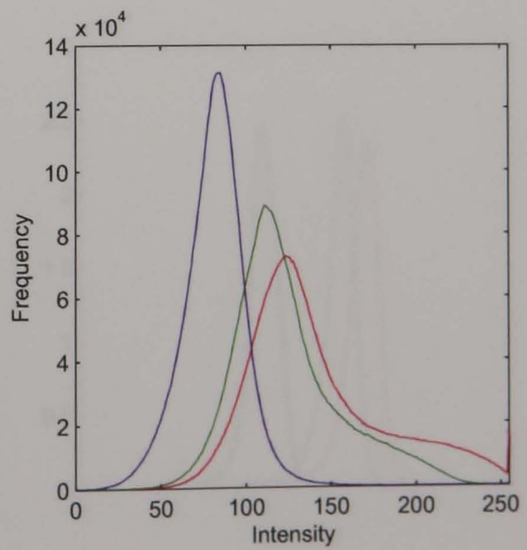


Figure G.7: road_02

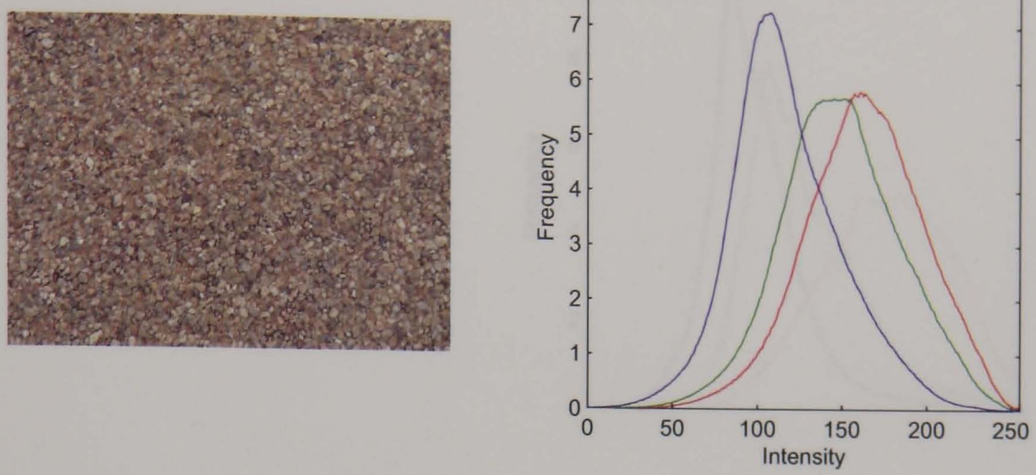


Figure G.8: road_03

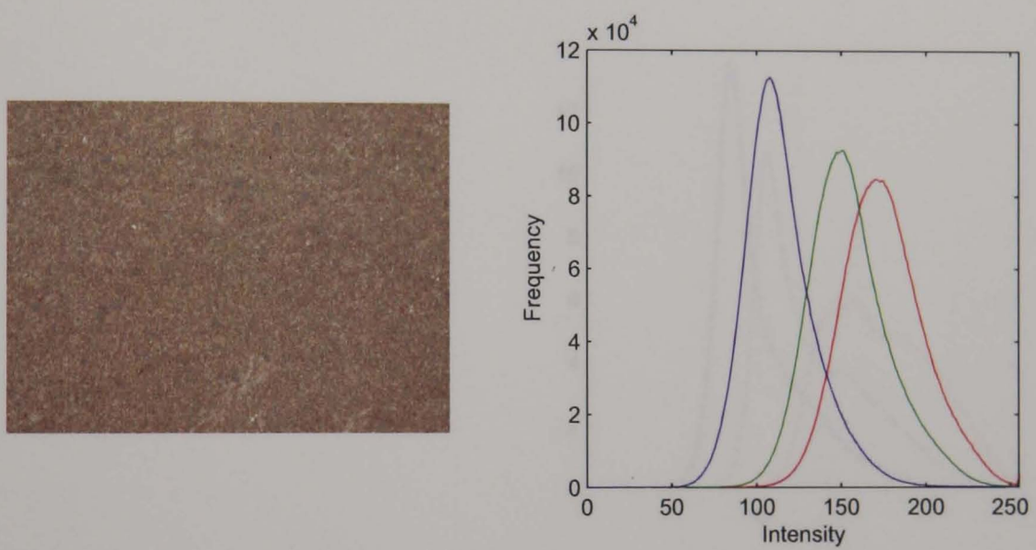


Figure G.9: stone_01

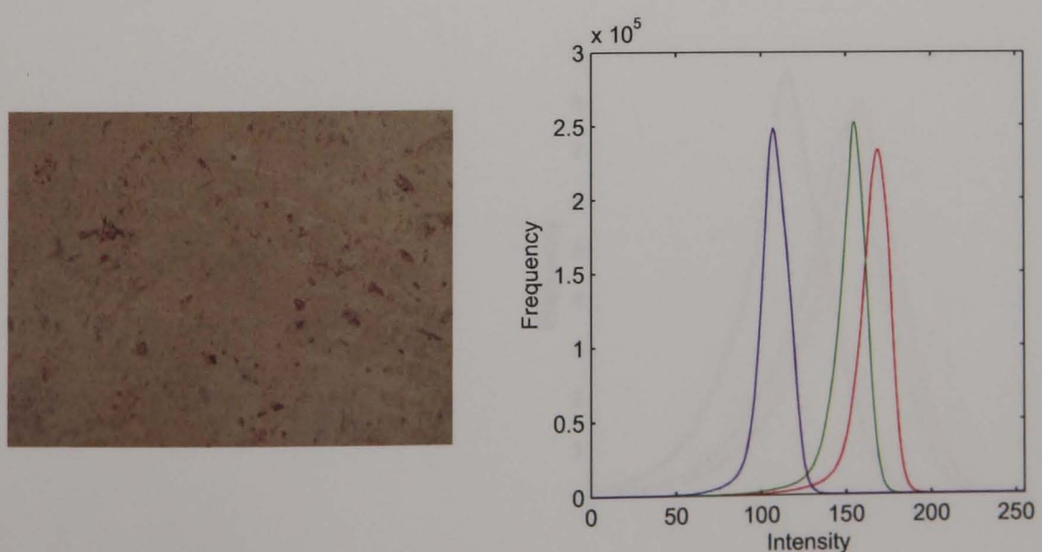


Figure G.10: stone_02

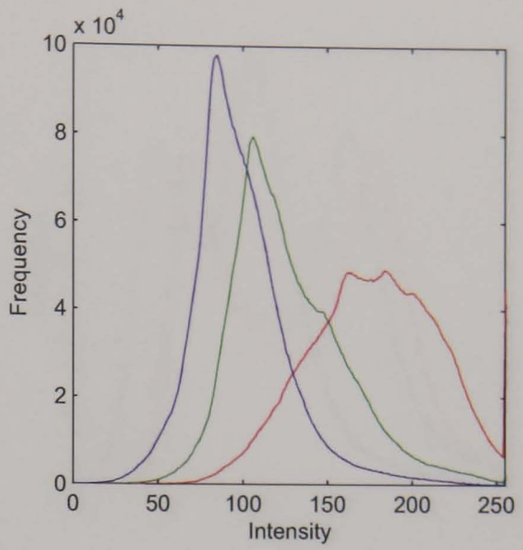


Figure G.11: stone_03

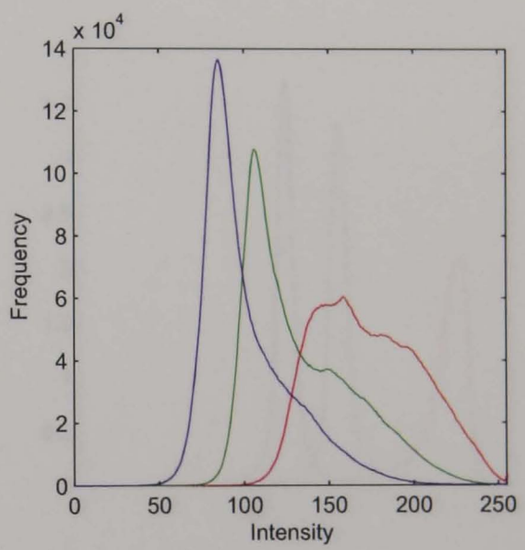


Figure G.12: stone_04

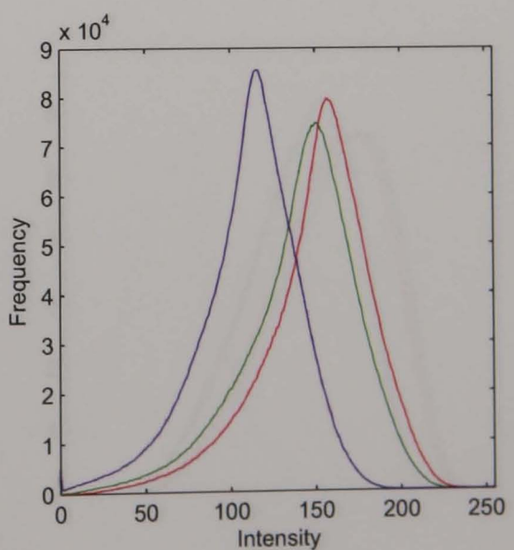
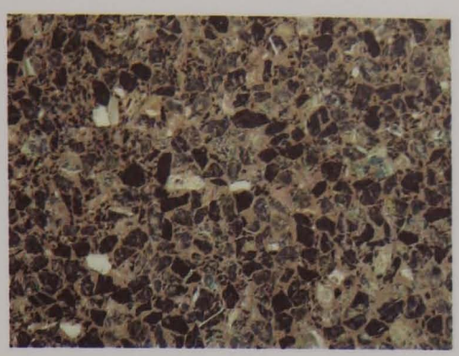


Figure G.13: stone_05



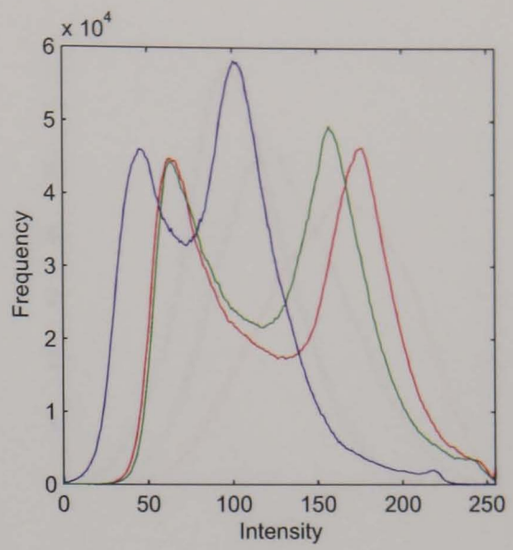
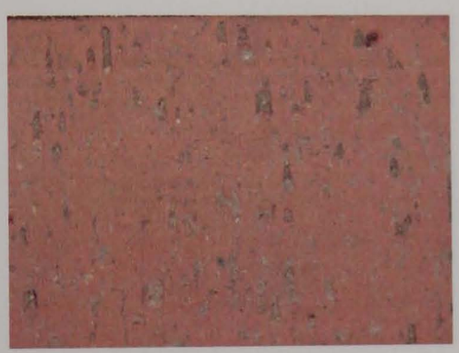


Figure G.14: stone_06



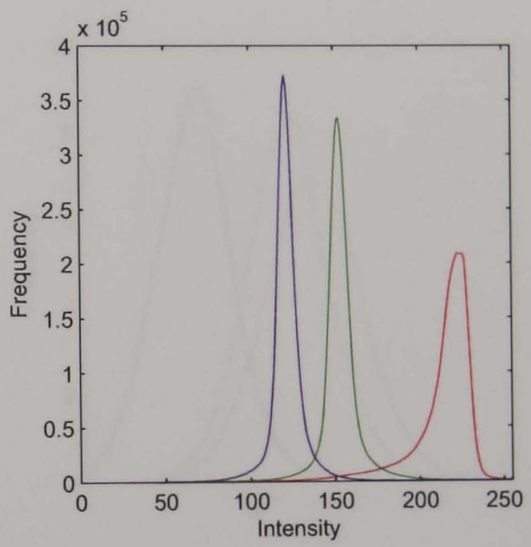


Figure G.15: stone_07

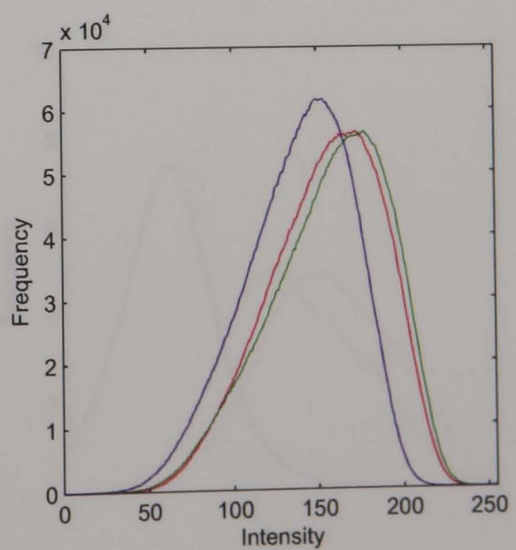


Figure G.16: stone_08

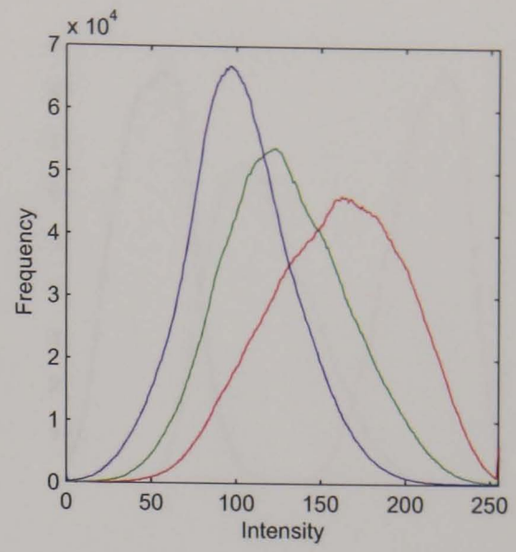


Figure G.17: stone_09

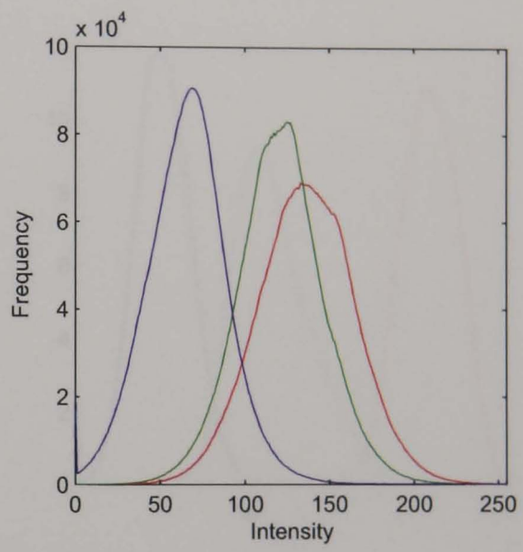


Figure G.18: tree_trunk_01

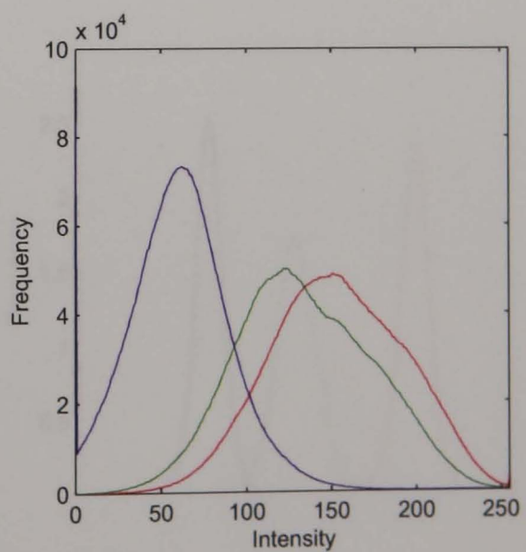


Figure G.19: tree_trunk_02

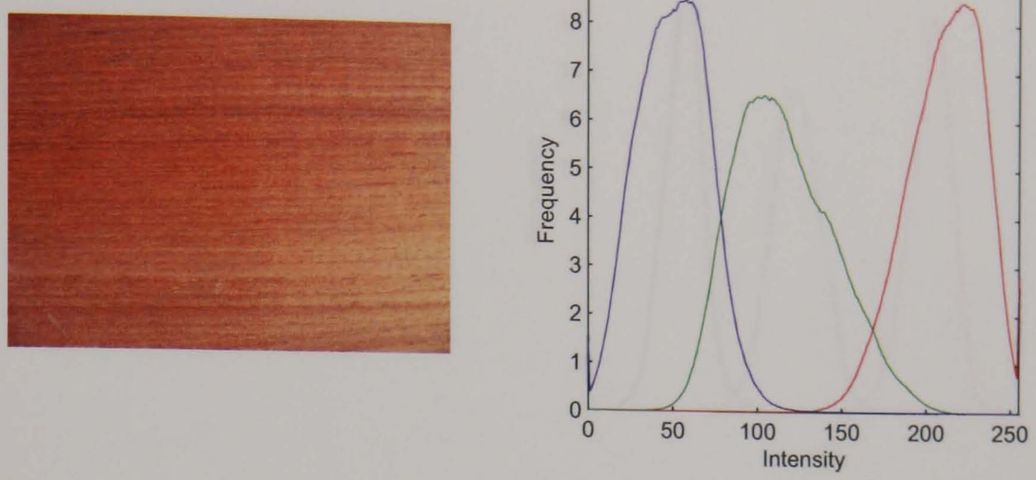


Figure G.20: wood_01

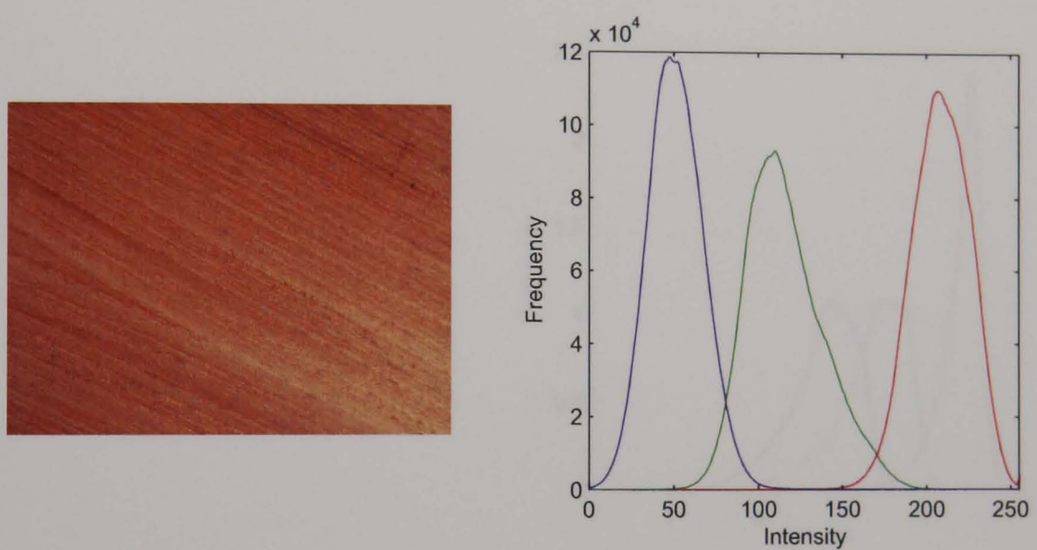


Figure G.21: wood_02

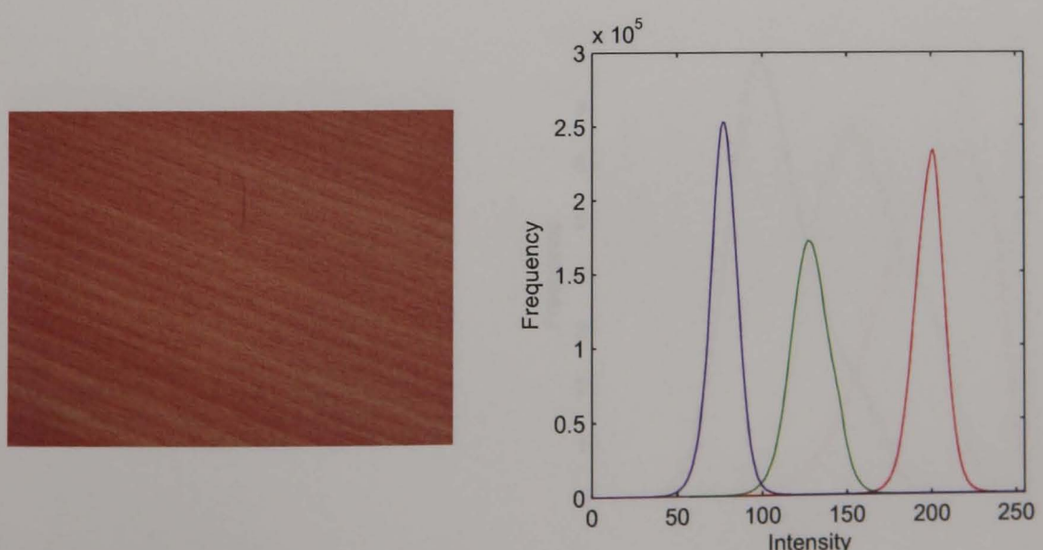


Figure G.22: wood_03

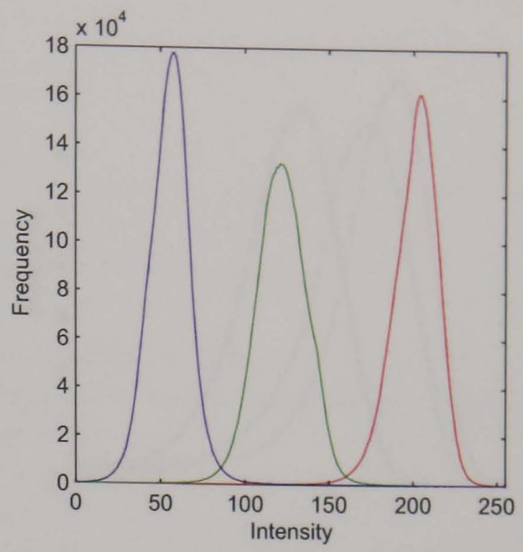


Figure G.23: wood_04

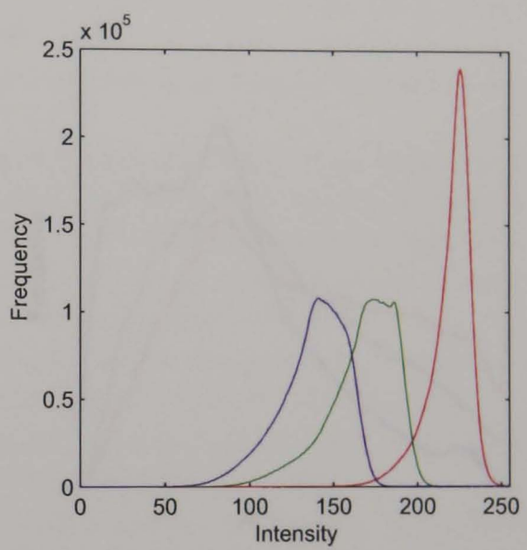
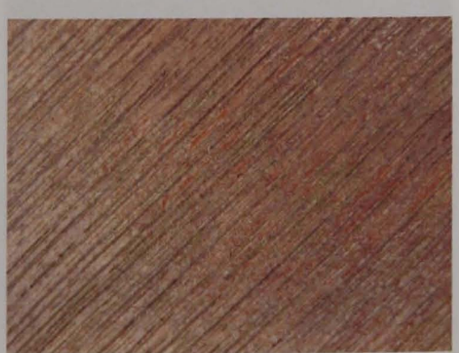


Figure G.24: wood_05



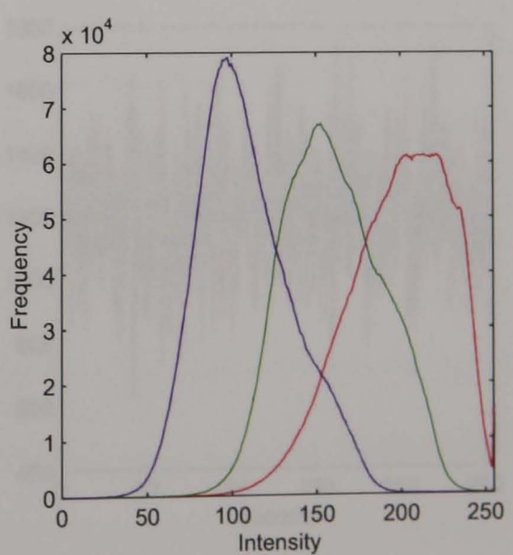


Figure G.25: wood_06

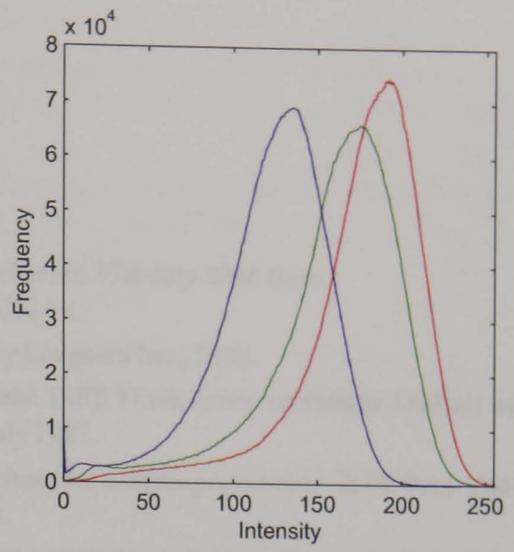


Figure G.26: wood_07

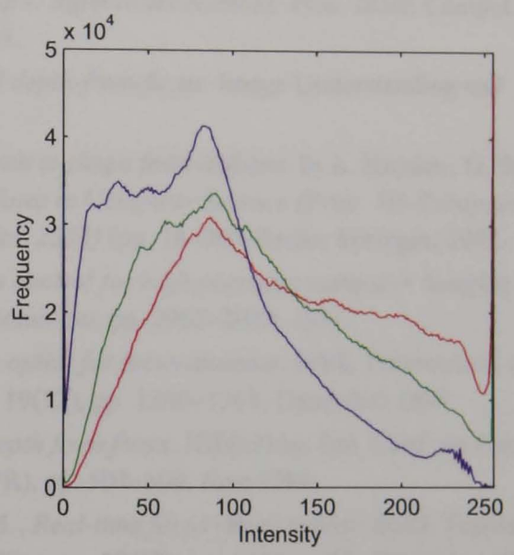


Figure G.27: wood_chips_01

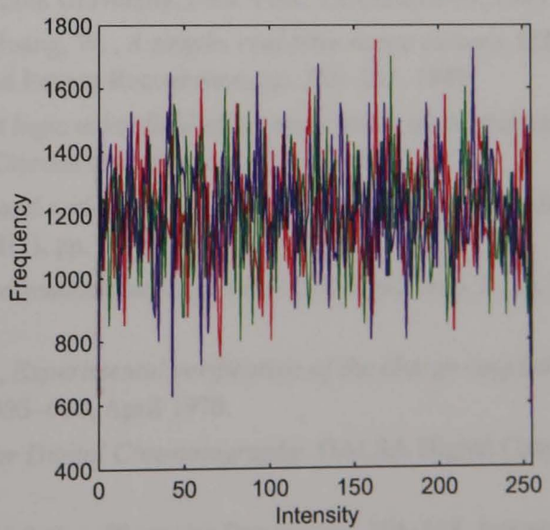


Figure G.28: Random colour checkerboard with 5×5 squares

References

[1] Hickie, W. J., *The Clouds (Translation)*. Retrieved 17th July 2006 from http://www.gutenberg.org/dirs/etext01/cloud10.txt

[2] Hecht, E., *Optics*. Reading: Addison Wesley Longman Inc., 2002.

[3] Pentland, A. P., *A new sense for depth of field*. IEEE Transactions on Pattern Analysis and Machine Intelligence, **9**(4), pp. 523–531, July 1987.

[4] Subbarao, M., *Parallel depth recovery by changing camera parameters*. IEEE Proc. 2nd Intl. Conf. Computer Vision, pp. 149–155, 1988.

[5] Horii, A., *Depth from defocusing*. Computational Vision and Active Perception Laboratory (CVAP), Royal Institute of Technology, Stockholm. ISRN KTH/NA/P--92/16--SE, 1992.

[6] Subbarao, M., *Direct recovery of depth-map I: differential methods*. Proc. IEEE Comput. Soc. Workshop Comput. Vision, pp. 58–65, 1987.

[7] Bove, V., *Discrete fourier transform based depth-from-focus*. Image Understanding and Machine Vision, pp. 118–121, 1989.

[8] Jin, H., & Favaro, P., *A variational approach to shape from defocus*. In A. Heyden, G. Sparr, M. Nielsen & P. Johansen (Ed.), *Lecture Notes in Computer Science (Proc. 7th European Conference on Computer Vision, Part II, Vol. 2351)* (pp. 18–30). Berlin: Springer, 2002.

[9] Willson, R. G., & Shafer, S. A., *Active lens control for high precision computer imaging*. IEEE Proc. Intl. Conf. on Robotics and Automation, pp. 2063–2070, 1991.

[10] Watanabe, M., & Nayar, S. K., *Telecentric optics for focus analysis*. IEEE Transactions on Pattern Analysis and Machine Intelligence, **19**(12), pp. 1360–1365, December 1997.

[11] Darrell, T., & Wohn, K., *Pyramid based depth from focus*. IEEE Proc. Intl. Conf. on Computer Vision and Pattern Recognition (CVPR), pp. 504–509, June 1988.

[12] Nayar, S. K., Watanabe, M., & Noguchi, M., *Real-time focus range sensor*. IEEE Transactions on Pattern Analysis and Machine Intelligence, **18**(12), pp. 1186–1198, December 1996.

[13] Watanabe, M., & Nayar, S. K., *Telecentric optics for constant-magnification imaging*. Department of Computer Science, Columbia University, New York. CUCS-026-95, 1995.

[14] Pentland, A., Darrell, T., Turk, M., & Huang, W., *A simple, real-time range camera*. IEEE Proc. Intl. Conf. on Computer Vision and Pattern Recognition, pp. 256–261, 1989.

[15] Wanlass, F. M., & Sah, C. T., *Nanowatt logic using field-effect metal-oxide semiconductor triodes*. IEEE International Solid-State Circuits Conference, pp. 32–33, 1963.

[16] Pelka, J. B., *Area detectors technology and optics -- relations to nature*. Nuclear Instruments and Methods in Physics Research A, **551**(1), pp. 52–65, 1 October 2005.

[17] Boyle, W., & Smith, G., *Charge coupled semiconductor devices*. Bell Syst. Tech. J., pp. 587–593, April 1970.

[18] Amelio, G., Tompsett, M., & Smith, G., *Experimental verification of the charge coupled device concept*. Bell Syst. Tech. J., pp. 593–600, April 1970.

[19] DALSA., *Image Sensor Architectures for Digital Cinematography*. DALSA Digital Cinema, Woodland Hills. 03-70-00218-01, 2002.

[20] Litwiller, D., *CCD vs. CMOS: Facts and fiction*. Photonics Spectra, pp. 154–158, January 2001.

[21] Janesick, J., *Dueling detectors*. OE Magazine, pp. 30–33, February 2002.

[22] Magnan, P., *Detection of visible photons in CCD and CMOS: A comparative review*. Nuclear Instruments and Methods in Physics Research A, **504**(1), pp. 199–212, 21 May 2003.

[23] Blanc, N., *CCD versus CMOS -- has CCD imaging come to an end?*. In D. Fritsch & R. Spiller (Ed.), *Photogrammetric Week '01* (pp. 131–137). Heidelberg: Wichmann Verlag, 2001.

[24] Hopkinson, G. R., Goodmand, T. M., & Prince, S. R., *A guide to the use and calibration of detector array equipment*. Sira: Sira Electro-Optics Limited, 2000.

[25] Andor., *CCD Sensor Architectures*. Retrieved 06/01/07 from http://www.andor.com/library/digital_cameras/?app=314

[26] Balser A631f User's Manual, Basler Vision Technologies

[27] Tanaka, J., Weiskopf, D., & Williams, P., *The role of color in high-level vision*. TRENDS in Cognitive Science, **5**(5), pp. 211–215, May 2001.

[28] Sharma, G., *Digital Color Imaging Handbook*. Boca Raton: CRC Press, 2003.

[29] Polyak, S. L., *The Retina*. Chicago: University of Chicago Press, 1941.

[30] Acharya, T., & Ray, A. K., *Image processing : principles and applications*. Hoboken: John Wiley, 2005.

[31] Wandell, B. A., Gamal, A. E., & Girod, B., *Common principles of image acquisition systems and biological vision* . Proceedings of the IEEE, **90**(1), pp. 5–17, January 2002.

[32] Kato, H., *Color reproduction test for CCD image sensors*. Proceedings of the International Test Conference, pp. 493–497, September 1990.

[33] Sony., *Datasheet for the ICX267AK CCD*. Retrieved 06/01/07 from http://www.ptgrey.com/support/kb/data/ICX267AK.pdf

[34] Plataniotis, K. N., & Venetsanopoulos, A. N., *Color Image Processing and Applications*. Berlin: Springer-Verlag, 2000.

[35] Berger, C. E., de Koeijer, J. A., Glas, W., & Madhuizen, H. T., *Color separation in forensic image processing*. Journal of Forensic Science, **51**(1), pp. 100–102, January 2006.

[36] Rabinovich, A., et al., *Unsupervised Color Decomposition of Histologically Stained Tissue Samples*. In S. Thrun (Ed.), *Proceedings of NIPS* (pp. 1–7). Cambridge: MIT Press, 2003.

[37] Horn, B., *Robot Vision*. Cambridge: MIT Press, 1986.

[38] Adelson, E. H., & Wang, J. Y., *Single lens stereo with a plenoptic camera*. IEEE Trans. Patt. Anal. and Mach. Intell., **14**(2), pp. 99–106, February 1992.

[39] Fox, J. S., *Range from translational motion blurring*. Proc. CVPR, pp. 360–365, 1988.

[40] Blackmore, S., *The Grand Illusion*. New Scientist, pp. 26–29, 22 June 2002.

[41] Chaudhuri, S., & Rajagopalan, A. N., *Depth from Defocus: A Real Aperture Approach*. New York: Springer, 1998.

[42] Subbarao, M., *Machine Vision for Inspection and Measurement*. Orlando: Academic Press, 1988.

[43] Nayar, S. K., *Shape from focus system for rough surfaces*. Proc. Image Understanding Workshop, pp. 593–606, 1992.

[44] Xu, S., Capson, D. W., & Caelli, T. M., *Range measurement from defocus gradient*. Machine Vision and Applications, pp. 179–186, 1995.

[45] Sonka, M., Hlavac, V., & Boyle, R., *Image Processing, Analysis, and Machine Vision*. Pacific Grove: PWS Publishing, 1999.

[46] Torralba, A., & Oliva, A., *Depth estimation from image structure*. IEEE Trans. Patt. Anal. and Mach. Intell., **24**(9), pp. 1226–1238, September 2002.

[47] Grau, O., & Thomas, G. A., *Use of image-based 3D modelling techniques in broadcast applications*. BBC R&D Department, Kingswood Warren. WHP 035, 2002.

[48] Swain, C., & Chen, T., *Defocus-based image segmentation*. IEEE International Conference on Acoustics, Speech and Signal Processing, **4**, pp. 2403–2406, 9th-12th May 1995.

[49] Price, M., et al., *Real-time production and delivery of 3D media*. BBC R&D, Surrey. WHP 045, 2002.

[50] Harman, P., *Home based 3D entertainment - an overview*. International Conference Proceedings on Image Processing, pp. 1–4, 2000.

[51] Jung, S., Park, J., & Lee, B., *Viewing-angle-enhanced integral 3-D imaging using double display devices with masks*. Optical Engineering, **41**(10), pp. 2389–2390, October 2002.

[52] Spitzer, T., *Missile Defense Technologies Tools to Counter Terrorism 2002*. Retrieved 4th September 2006 from http://www.mda.mil/mdalink/pdf/terror.pdf

[53] Soper, N. J., Odem, R. R., Clayman, R. V., & McDoughall, E. M., *Essentials of Laparoscopy*. St. Louis: Quality Medical Publishing, Inc., 1994.

[54] Marescaux, J., et al., *Transatlantic robot-assisted telesurgery*. Nature, **413**, pp. 379–380. 27 September 2001.

[55] Reichenbach, S. E., Park, S. K., & Narayanswamy, R., *Characterizing digital image acquisition devices*. Optical Engineering, **30**(2), pp. 170–177, February 1991.

[56] Tzannes, A. P., & Mooney, J. M., *Measurement of the modulation transfer function of infrared cameras*. Optical Engineering, **34**(6), pp. 1808–1817, June 1995.

[57] Staunton, R. C., *Edge operator error estimation incorporating measurements of CCD TV camera transfer function*. IEE Proc.-Vis. Signal Process., **145**(3), June 1998.

[58] Ens, J., & Lawrence, P., *A matrix based method for determining depth from focus*. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 600–606, 3rd-6th June 1991.

[59] Ens, J., & Lawrence, P., *An investigation of methods for determining depth from focus*. IEEE Trans. Patt. Anal. and Mach. Intell., **15**(2), pp. 97–107, February 1993.

[60] Pentland, A., *Depth of scene from depth of field*. Proc. Image Understanding Workshop, pp. 253–259, September 1982.

[61] Kosara, R., Miksch, S., & Hauser, H., *Focus and context taken literally*. IEEE Computer Graphics and Applications, **22**(1), pp. 22–29, Jan.-Feb. 2002.

[62] Garcia, J., Sanchez, J. M., Orriols, X., & Binefa, X., *Chromatic aberration and depth extraction*. IEEE Proc. 15th Intl. Conf. Pattern Recognition, **1**, pp. 762–765, 3-7 September 2000.

[63] Fincham, E., *The accomodation reflex and its stimulus*. British Journal of Ophtalmology, pp. 381–393, 1951.

[64] Weale, R. A., *Focus on Vision*. Cambridge: Harvard University Press, 1982.

[65] Schechner, Y. Y., & Kiryati, N., *Depth from defocus vs. stereo: how different really are they?*. Proc. Intl. Conf. on Pattern Recognition, pp. 1784–1786, 1998.

[66] von Helmholtz, H., *Helmholtz's Treatise on Physiological Optics*. Rochester: Opt. Soc. Amer., 1924.

[67] Lai, S. H., Fu, C. W., & Chang, S., *A generalized depth estimation algorithm with a single image*. IEEE Trans. Pattern Anal. and Mach. Intell., pp. 405–411, 1992.

[68] Subbarao, M., & Gurumoorthy, N., *Depth recovery from blurred edges*. Proc. of IEEE International Conference on Computer Vision and Pattern Recognition, pp. 498–503, 5th-9th June 1988.

[69] Saadat, A., *A simple general and mathematically tractable way to sense depth in a single image*. Proc. SPIE, Applications of Digital Image Processing XVIII, pp. 355–363, August 1995.

[70] Simon, C., Bicking, F., & Simon, T., *Estimation of depth on thick edges from sharp and blurred images*. Proc. 19th IEEE Instrumentation and Measurement Technology Conf., **1**, pp. 323–328, 21-23 May 2002.

[71] Simon, C., Bicking, F., & Simon, T., *Depth estimation based on thick orientated edges in images*. IEEE International Symposium on Industrial Electronics, 2004 , pp. 135–140, May 2004.

[72] Tsai, D., & Lin, C., *A moment-preserving approach for depth from defocus*. Pattern Recognition, pp. 551–560, 1998.

[73] Rajagopalan, A. N., & Chaudhuri, S., *Optimal selection of camera parameters for recovery of depth from defocused images*. Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 219–224, 17th-19th June 1997.

[74] Watanabe, M., & Nayar, S. K., *Telecentric optics for computational vision*. In B. F. Buxton (Ed.), *Lecture Notes In Computer Science, Proceedings of the 4th European Conference on Computer Vision-Volume II* (pp. 439–451). London: Springer-Verlag, 1996.

[75] Hwang, T., Clark, J. J., & Yuille, A. L., *A depth recovery algorithm using defocus information*. IEEE Proc. on Computer Vision and Pattern Recognition, pp. 476–482, 4th-8th June 1989.

[76] Pentland, A., Scherock, S., Darrell, T., & Girod, B., *Simple range cameras based on focal error*. J. Opt. Soc. Am. A, **11**(4), pp. 2925–2934, November 1994.

[77] Xiong, Y., & Shafer, S. A., *Depth from focus and defocusing*. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 68–73, June 1993.

[78] Subbarao, M., & Wei, T., *Depth from defocus and rapid autofocusing: a practical approach*. Proc. CVPR, pp. 773–776, 1992.

[79] Surya, G., & Subbarao, M., *Depth from defocus by changing camera aperture: a spatial domain approach*. Proc. CVPR, pp. 61–67, 1993.

[80] Subbarao, M., & Surya, G., *Application of spatial-domain convolution / deconvolution transform for determining distance from image defocus*. Proc. SPIE, Optics, Illumination, and Image Sensing for Machine Vision VII, pp. 159–167, November 1992.

[81] Watanabe, M., & Nayar, S. K., *Rational filters for passive depth from defocus*. Intl. J. Computer Vision, **27**(3), pp. 203–225, May 1998.

[82] Subbarao, M., & Surya, G., *Depth from defocus: a spatial domain approach*. Intl. J. of Computer Vision, pp. 271–294, 1994.

[83] Yuan, T., & Subbarao, M., *Integration of multiple-baseline color stereo vision with focus and defocus analysis for 3D shape measurement*. Proc. SPIE, Three Dimensional Imaging, Optical Metrology, and Inspection IV, pp. 44–51, December 1998.

[84] Subbarao, M., Wei, T., & Surya, G., *Focused image recovery from two defocused images recorded with different camera settings*. IEEE. Trans. on Image Processing, **4**(12), pp. 1613–1627, December 1995.

[85] Ziou, D., *Passive depth from defocus using a spatial domain approach*. Proc. 6th Intl. Conf. Computer Vision , pp. 799–804, 4-7 January 1998.

[86] Ziou, D., & Deschenes, F., *Depth from defocus estimation in spatial domain*. Computer Vision and Image Understanding, pp. 143–165, 2001.

[87] Rayala, J., Gupta, S., & Mullick, S. K., *Estimation of depth from defocus as polynomial system identification*. IEE Proc. Vis. Image Signal Process., **148**(5), pp. 356–362, October 2001.

[88] Deschênes, F., Ziou, D., & Fuchs, P., *Enhanced depth from defocus estimation: tolerance to spatial displacements*. Proc. Intl. Conf. Image and Signal Processing, **2**, pp. 978–985, 2-5 May 2001.

[89] Farid, H., & Simoncelli, E. P., *Range estimation by optical differentiation*. Journal of the Optical Society of America A, **15**(7), pp. 1777–1786, July 1988.

[90] Xiong, Y., & Shafer, S. A., *Moment and hypergeometric filters for high precision computation of focus, stereo and optical flow*. International Journal of Computer Vision, **22**(1), pp. 25–59, February 1997.

[91] Nayar, S. K., Watanabe, M., & Noguchi, N., *Real-time focus range sensor*. Columbia University, New York. CUCS-028-94, 1994.

[92] Rajagopalan, A. N., & Chaudhuri, S., *Space-variant approaches to recovery of depth from defocused images*. Computer Vision and Image Understanding, **68**(3), pp. 309–329, December 1997.

[93] Rajagopalan, A. N., & Chaudhuri, S., *Optimal recovery of depth from defocused images using an MRF model*. Procedings of the Sixth International Conference on Computer Vision, pp. 1047–1052, 4th-7th January 1998.

[94] Favaro, P., & Soatto, S., *A geometric approach to shape from defocus*. IEEE. Trans. Pattern Analysis and Machine Intelligence, **27**(3), pp. 406–417, March 2005.

[95] Rajan, D., Chaudhuri, S., & Joshi, M. V., *Multi-objective super resolution: concepts and examples*. IEEE Signal Processing Magazine, **20**(3), pp. 49–61, May 2003.

[96] Favaro, P., & Soatto, S., *Shape and radiance estimation from the information-divergence of blurred images*. In D. Vernon (Ed.), *Lecture Notes in Computer Science (Proceedings of the 6th European Conference on Computer Vision-Part I)* (pp. 755–768). London: Springer-Verlag, 2000.

[97] Csiszár, I., *Why least squares and maximum entropy; an axiomatic approach to inverse problems*. Annals of Statistics, pp. 2033–2066, 1991.

[98] Favaro, P., & Soatto, S., *Learning shape from defocus*. In A. Heyden, G. Sparr, M. Nielsen & P. Johansen (Ed.), *Lecture Notes in Computer Science (Proc. 7th European Conference on Computer Vision, Part II, Vol. 2351)* (pp. 735–745). Berlin: Springer-Verlag, 2002.

[99] Prasad, K. V., & Mammone, R. J., *Depth restoration from defocused images using simulated annealing*. Proc. 10th Intl. IEEE Conf. on Pattern Recognition, **1**, pp. 227–229, 16th-21st June 1990.

[100] Bove, V., *Entropy-based depth from focus*. Journal Optical Society of America A, **10**(4), pp. 561–566, April 1993.

[101] Swain, C., Peters, A., & Kawamura, K., *Depth estimation from image defocus using fuzzy logic*. Proceedings of the IEEE International Conference on Fuzzy Systems, pp. 94–99, June 1994.

[102] Kim, B., Yun, J., & Choi, T. S., *Shape recovery from blurred image using wavelet analysis*. Proc. SPIE, Digital Image Recovery and Synthesis IV, pp. 248–258, July 1999.

[103] Hor, M., Chen, J., & Chen, K., *Wavelet transform in depth recovery*. Proc. SPIE, Intelligent Robots and Computer Vision XII: Algorithms and Techniques, pp. 463–474, August 1993.

[104] Hiura, S., & Matsuyama, T., *Depth measurement by the multi-focus camera*. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 953–959, June 1998.

[105] Murata, S., & Kawamura, M., *Particle depth measurement based on depth-from-defocus*. Optics and Laser Technology, **31**(8), pp. 95–102, June 1999.

[106] Ghita, O. G., & Whelan, P., *Real time 3-D estimation using depth from defocus*. Proc. Irish Machine Vision and Image Processing Conference, pp. 167–181, 8-9 September 1999.

[107] Ghita, O., & Whelan, P. F., *A video-rate range sensor based on depth from defocus*. Optics & Laser Technologuly, **33**(3), pp. 167–176, April 2001.

[108] Ma, L., & Staunton, R., *Integration of multiresolution image segmentation and neural networks for object depth recovery*. Pattern Recognition, pp. 985–996, 2005.

[109] Abbott, A. L., & Ahuja, N., *Surface reconstruction by dynamic integration of focus, camera vergence, and stereo*. Second Intl. Conf. on Computer Vision, pp. 532–543, December 1988.

[110] Yadid-Pecht, O., *Geometrical modulation transfer function for different pixel active area shapes*. Optical Engineeringq, **39**(4), pp. 859–865, April 2000.

[111] Healey, G. E., & Kondepudy, R., *Radiometric CCD camera calibration and noise estimation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, **16**(3), pp. 267–276, March 1994.

[112] Kavaldjiev, D., & Ninkov, Z., *Influence of nonuniform charge-coupled device pixel response on aperture photometry*. Optical Engineering, **40**(2), pp. 162–169, February 2001.

[113] Lauer, T. R., *The photometry of undersampled point-spread functions*. The Publications of the Astronomical Society of the Pacific, **111**(765), pp. 1434–1443, November 1999.

[114] Feltz, J. C., & Karim, M. A., *Modulation transfer function of charge-coupled devices*. Applied Optics, **29**(5), pp. 717–722, 10 February 1990.

[115] Marchywka, M., & Socker, D. G., *Modulation transfer function measurement technique for small-pixel detectors*. Applied Optics, SPIE, **31**(34), pp. 7198–7213, 1 December 1992.

[116] Hu, J., Song, M., Sun, Y., & Li, Y., *Measurement of modulation transfer function of charge-coupled devices using frequency-variable sine grating patterns*. Optical Engineering, **38**(7), pp. 1200–1204, July 1999.

[117] Boreman, G., & Dereniak, E. L., *Method for measuring modulation transfer function of charge-coupled device using laser speckle*. Optical Engineering, pp. 148–150, 1986.

[118] Sensiper, M., Boreman, G. D., & Ducharme, A. D., *Modulation transfer function testing of detector arrays using narrow-band laser speckle*. Optical Engineering, **32**(2), pp. 395–400, February 1993.

[119] Boreman, G. D., Sun, Y., & James, A. B., *Generation of laser speckle with an integrating sphere*. Optical Engineering, **29**(4), pp. 339–342, April 1990.

[120] Ducharme, A. D., & Boreman, G. D., *Holographic elements for modulation transfer function testing of detector arrays*. Optical Engineering, **34**(8), pp. 2455–2458, August 1995.

[121] Daniels, A., Boreman, G. D., & Ducharme, A. D., *Random transparency targets for modulation transfer function measurement in the visible and infrared regions*. Optical Engineering, **34**(3), pp. 860–868, March 1995.

[122] Reimann, D. A., Jacobs, H. A., & Samei, E., *Use of Weiner filtering in the measurement of the two-dimensional modulation transfer function*. Proceedings of the SPIE, Physics of Medical Imaging, pp. 670–680, 2000.

[123] Schulz, T. J., *Multiframe image restoration*. In A. Bovik (Ed.), *Handbook of image & video processing* (pp. 175–189). San Diego: Academic Press, 2000.

[124] Parker, G. J., *Introductory Semiconductor Device Physics*. Hemel Hempstead: Prentice Hall Europe (UK) Limited, 1994.

[125] Mathews, J. H., *Computer derivations of numerical differentiation formulae*. Int. J. of Math. Education in Sci. and Tech., **34**(2), pp. 280–287, March 2003.

[126] Chartrand, R., *Numerical differentiation of noisy, nonsmooth data*. Submitted, 2005. http://math.lanl.gov/Research/Publications/Docs/chartrand-2005-numerical.pdf

[127] Gonzalez, R. C., & Woods, R. E., *Digital Image Processing*. Upper Saddle River: Prentice-Hall, Inc., 2002.

[128] Zeidler, E., *Oxford Users' Guide to Mathematics*. New York: Oxford University Press Inc., 2004.

[129] Alleysson, D., Susstrunk, S., & Herault, J., *Linear demosaicing inspired by the human visual system*. IEEE Transactions on Image Processing, **14**(4), pp. 439–449, April 2005.

[130] Kimmel, R., *Demosaicing: image reconstruction from color CCD samples*. IEEE Transactions on Image Processing, **8**(9), pp. 1221–1228, September 1999.

[131] Farsiu, S., Elad, M., & Milanfar, P., *Multiframe demosaicing and super-resolution of color images*. IEEE Transactions on Image Processing, **15**(1), pp. 141–159, January 2006.

[132] Ray, S. F., *Applied Photographic Optics: Imaging Systems for Photography, Film and Video.* London: Focal Press, 1988.

[133] Kullback, S., *Information Theory and Statistics.* New York: Wiley, 1959.

[134] Bose, T., *Digital Signal and Image Processing.* Hoboken: John Wiley & Sons, Inc., 2004.

[135] Forsyth, D. A., & Ponce, J., *Computer Vision: A Modern Approach.* Upper Saddle River: Prentice-Hall, Inc., 2003.

[136] Laney, D., *Lens practice: Choosing and using Leica lenes.* East Sussex: Hove Books, 1993.

[137] Man, K. F., Tang, K. S., & Kwong, S., *Genetic Algorithms: Concepts and Designs.* London: Springer-Verlag London Limited, 2001.

[138] Pearson, K., *On lines and planes of closest fit to systems of points in space.* Phil. Mag., pp. 559–572, 1901.

[139] Hotelling, H., *Analysis of a complex of statistical variables into principal components* . J. Educ. Psychol., pp. 417–441, 1933.

[140] Duda, R. O., Hart, P. E., & Stork, D. G., *Pattern Classification (2nd edition).* Hoboken: John Wiley & Sons, Inc., 2001.

[141] Petrou, M., & Bosdogianni, P., *Image Processing: The Fundamentals.* Chichester: John Wiley & Sons, Ltd., 1999.

[142] Jolliffe, I. T., *Principal Component Analysis.* New York: Springer-Verlag New York, Inc., 2002.

[143] DeGroot, M. H., *Probability and Statistics.* Reading: Addison-Wesley, 1989.

[144] Tuceryan, M., & Jain, A. K., *Texture Analysis.* In C. H. Chen, L. F. Pau & P. S. Wang (Ed.), *The Handbook of Pattern Recognition and Computer Vision (2nd Edition)* (pp. 207–248). River Edge: World Scientific Publishing Co., Inc., 1998.

[145] Laws, K. I., *Textured image segmentation*, Doctoral Dissertation, University of Southern California, Los Angeles, 1980.

[146] Muneeswaran, K., Ganesan, L., Arumugam, S., & Soundar, K. R., *Texture classification with combined rotation and scale invariant wavelet features.* Pattern Recognition, pp. 1495–1506, 2005.

[147] Haralick, R. M., & Shapiro, L. G., *Computer and Robot Vision.* Reading: Addison-Wesley Publishing Company, Inc., 1992.

[148] Bharati, M. H., & MacGregor, J. F., *Texture analysis of images using principal component analysis.* Proceedings of the SPIE, Process Imaging for Automatic Control, pp. 27–37, 2001.

[149] Julesz, B., Gilbert, E. N., & Victor, J. D., *Visual discrimination of textures with identical third order statistics.* Biological Cybernetics, pp. 137–140, 1978.

[150] Tou, J. T., & Chang, Y. S., *Picture understanding by machine via textural feature extraction.* Proceedings of the IEEE Conference on Pattern Recognition and Image Processing, pp. 392–299, 1977.

[151] McCormick, B. H., & Jayaramamurthy, S. N., *Time series models for texture synthesis.* Journal of Computer and Information Sciences, pp. 329–343, 1971.

[152] Schacter, B., Rosenfeld, A., & Davis, L. S., *Random mosaic models for textures.* IEEE Transactions on Systems, Man and Cybernetics, pp. 694–702, 1978.

[153] Pentland, A., *Fractal-based description of natural scenes.* IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 661–674, 1984.

[154] Fazel-Rezai, R., & Kinsner, W., *Texture analysis and segmentation of images using fractals.* Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering, **2**, pp. 786–791, 9-12 May 1999.

[155] Liu, X., & Wang, D., *Texture classification using spectral histograms.* IEEE Transactions on Image Processing, **12**(6), pp. 661–670, June 2003.

[156] Laws, K. I., *Rapid texture identification*. Proceedings of the SPIE Conference on Image Processing for Missile Guidance, pp. 376–380, August 1980.

[157] Artmann, B., *Euclid - The Creation of Mathematics*. New York: Springer-Verlag New York Inc., 1991.

[158] Mandelbrot, B. B., *Fractals: Form, Chance, and Dimension*. San Francisco: W. H. Freeman, 1977.

[159] Voss, R. J., *Fractals in Nature: from Characterization to Simulation*. In H. Peitgen & D. Saupe (Ed.), *The Science of Fractal Images* (pp. 21–70). New York: Springer-Verlag, 1988.

[160] Mandelbrot, B. B., *How long is the coastline of Britain? Statistical self-similarity and fractional dimension*. Science, pp. 636–638, 1967.

[161] Turner, M. J., Blackledge, J. M., & Andrews, P. R., *Fractal Geometry in Digital Imaging*. London: Academic Press, 1998.

[162] Takayasu, H., *Fractals in the Physical Sciences*. Manchester: Manchester University Press, 1990.

[163] Romeu, D., et al., *Surface fractal dimension of small metallic particles*. Phys. Rev. Lett., **57**(20), pp. 2552–2555, 17 November 1986.

[164] Pathirana, A., & Herath, S., *Multifractal modelling and simulation of rain fields exhibiting spatial heterogeneity*. Hydrology and Earth System Sciences, pp. 695–708, 2002.

[165] Cochran, W. O., Hart, J. C., & Flynn, P. J., *On approximating rough curves with fractal functions*. Proceedings of Graphics Interface, pp. 65–72, 1998.

[166] Fox, C. G., & Hayes, D. E., *Quantitative methods for analyzing the roughness of the seafloor*. Reviews of Geophysics and Space Physics, pp. 1–48, 1985.

[167] Jalobeanu, A., *Fractal 3-D modeling of asteroids using wavelets on arbitrary meshes*. Proc. of IAFA, Bucharest, Romania, pp. 1–6, May 2003.

[168] Barthel, K. U., & Cycon, H. L., *Image denoising using fractal and wavlet-based methods*. Proceedings of the SPIE, Wavelet Applications in Industrial Processing, pp. 39–47, February 2004.

[169] Furlan, W. D., Saavedra, G., Monsoriu, J. A., & Patrignani, J. D., *Axial behaviour of Cantor ring diffractals*. Journal of Optics A: Pure and Applied Optics, pp. 361–364, 2003.

[170] Avnir, D., Biham, O., Lidar, D., & Malcai, O., *Is the geometry of nature fractal?*. Science, **279**, pp. 39–40, 2 January 1998.

[171] Keller, J. M., Chen, S., & Crownover, R. M., *Texture description and segmentation through fractal geometry*. Computer Vision, Graphics, and Image Processing, pp. 150–166, 1989.

[172] Power, W. L., & Tullis, T. E., *Euclidean and fractal models for the description of rock surface roughness*. Journal of Geophysical Research, pp. 415–424, 1991.

[173] Klinkenberg, B., *A review of methods used to determine the fractal dimension of linear features*. Mathematical Geology, **26**(1), pp. 23–46, January 1994.

[174] Dubuc, B., et al., *Evaluating the fractal dimension of profiles*. Physical Review A (General Physics), **39**(3), pp. 1500–1512, February 1989.

[175] Liu, J., Hwang, W., & Chen, M., *Estimation of 2-D noisy fractional Brownian motion and its applications using wavelets*. IEEE Transactions on Image Processing, **9**(8), pp. 1407–1419, August 2000.

[176] Penn, A. I., & Loew, M. H., *Estimating fractal dimension with fractal interpolation function models*. IEEE Transactions on Medical Imaging, **16**(6), pp. 930–937, December 1997.

[177] Kress, R., *Linear Integral Equations*. Berlin: Springer-Verlag Inc., 1989.

[178] Tarantola, A., *Inverse problem theory: Methods for data fitting and model parameter estimation*. Amsterdam: Elsevier Science Publishers B.V., 1987.

[179] Demmel, J. W., *The probability that a numerical analysis problem is difficult*. Mathematics of Computation, **50**(182), pp. 449–480, April 1988.

[180] Edelman, A., *Eigenvalues and condition numbers of random matrices*, Doctoral Dissertation, Massachusetts Institute of Technology, Cambridge, 1989.

[181] Chen, Z., & Dongarra, J. J., *Condition numbers of Gaussian random matrices*. SIAM Journal on Matrix Analysis and Applications, pp. 603–620, 2005.

[182] Sontag, E. D., *Mathematical control theory: deterministic finite dimensional systems*. New York: Springer-Verlag New York, Inc., 1998.

[183] Asada, N., Fujiwara, H., & Matsuyama, T., *Seeing behind the scene: Analysis of photometric properties of occluding edges by the reversed projection blurring model*. IEEE Transactions on Pattern Analysis and Machine Intelligence, **20**(2), pp. 155–167, February 1998.

[184] Corner, B. R., Narayanan, R. M., & Reichenbach, S. E., *Principal component analysis of remote sensing imagery: effects of additive and multiplicative noise*. Proc. SPIE, Applications of Digital Image Processing XXII, pp. 183–191, October 1999.

[185] Boncelet, C., *Image Noise Models*. In A. Bovik (Ed.), *Handbook of Image and Video Processing* (pp. 325–336). San Diego: Academic Press, 2000.

[186] Rosipal, R., Girolami, M., Trejo, L. J., & Cichocki, A., *Kernel PCA for feature extraction and de-noising in nonlinear regression*. Neural Computing & Applications, pp. 231–243, 2001.

[187] Green, A. A., Berman, M., Switzer, P., & Craig, M. D., *A transformation for ordering multispectral data in terms of image quality with implications for noise removal*. IEEE Transactions on Geoscience and Remote Sensing, pp. 65–74, January 1988.

[188] Withagen, P. J., Groen, F. C., & Schutte, K., *CCD characterization for a range of color cameras*. Proceedings of the IEEE Instrumentation and Measurement Technology Conference, Ottawa, Ontario, Canada, **3**, pp. 2232–2235 , 17th - 19th May 2005.

[189] Eriksson, L., Johansson, E., & Wikström, C., *Tutorial. Mixture design -- design generation, PLS analysis, and model usage*. Chemometrics and Intelligent Laboratory Systems, pp. 1–24, 1998.

[190] Cornell, J. A., *Experiments with Mixtures*. Chichester: John Wiley & Sons, Inc., 2002.

[191] Chasalow, S. D., & Brand, R. J., *Generation of simplex lattice points*. Journal of Applied Statistics, pp. 534–545, 1995.